Multi-scale texture segmentation of synthetic aperture radar images

Fletcher, Neil David

Award date:
2005

Awarding institution:
University of Bath

MULTI-SCALE TEXTURE SEGMENTATION OF SYNTHETIC APERTURE RADAR IMAGES

Submitted by Neil David Fletcher
for the degree of
Doctor of Philosophy
of the University of Bath
2005

ProQuest LLC
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106-1346

Abstract

Texture analysis of synthetic aperture radar (SAR) images is a complex problem. One of the reasons for this is the absence of a standard measure for describing image texture. Existing texture measures are often based on the statistical properties of images in the spatial and spectral domains. As SAR is a coherent imaging system, its images are characterised by multiplicative noise known as speckle which adversely affects textural measures. Multi-scale analysis has been demonstrated to be an effective approach to the classification and segmentation of images and this thesis focuses on the application of multi-scale textural segmentation techniques to SAR images.

Traditional linear segmentation methods using Gabor and wavelet filterbanks are studied and a new wavelet packet segmentation approach is presented and evaluated on both synthetic and SAR images. In addition, the adverse effects of using the non-redundant wavelet representation are shown to be negligible. Non-linear segmentation operators based on Mathematical Morphology are also investigated. The greyscale local area granulometry technique is proposed as a new approach to morphological segmentation that does not suffer from any bias introduced by fixed structuring elements. Experimental results show the performance of this new approach to exceed that of existing local granulometry methods.

A further feature of this work has been to demonstrate the benefits of incorporating speckle reduction techniques within the segmentation methods. Wavelet shrinkage is an established technique for speckle reduction. Here, a new morphological speckle reduction method termed granulometric shrinkage is presented, that is shown to offer a flexible approach to speckle reduction with texture preservation.

Acknowledgements

The author would like to thank Dr. A.N. Evans of the University of Bath whose supervision and suggestions have guided the research. I should also mention the members of the Signal and Image Processing Group whose numerous discussions, about everything from Linux to LaTeX, have proved invaluable.

Thanks must also go to QinetiQ for their additional funding of this project and for supplying radar data. With special thanks to Dr. M.A. Evans of the Radar Image Systems Group at QinetiQ for initial guidance to the subject of texture segmentation.

Finally I would like to thank my parents and Carrie, without whose support, this research would not have been possible.

Contents

List of Figures

List of Tables

Statement of Originality

The author considers the following elements of this work form an original contribution to texture analysis and speckle reduction of synthetic aperture radar imagery literature.

Chapter 3: Through researching existing techniques for the suppression of speckle noise in SAR imagery the novel *granulometric shrinkage* algorithm has been developed [40]. Speckle reduction via granulometric shrinkage is achieved by applying a thresholding function to the grain images of a morphological data sieve. Granulometric shrinkage has the property of being able to reduce the noise components of a SAR image whilst preserving small scale image features.

By identifying which regions within an image contain features/textural information, it is possible to obtain a balance between speckle suppression and detail preservation. This is achieved by apply an adaptive threshold value to the image grain responses leading to global optimum speckle reduction results. This is an area of current research.

Chapter 5: Existing wavelet based texture segmentation methods have focused on the wavelet pyramid transform *i.e.* the recursive decomposition of the low frequency subband of an image. However, it is well established that textural information resides in mid-frequency regions of an image, therefore it is information extracted from these regions that are most applicable to texture segmentation. In this chapter a new wavelet-based technique for segmenting SAR images has been developed that forms its feature vectors from a wavelet packet transform [39].

By defining a feature space that allows for the separation of image feature components and speckle noise components, it has been possible reduce the effect

of speckle noise on the texture features.The incorporation of speckle reduction into a texture feature extraction framework leads to a more robust segmentation algorithm with less classification errors.

Chapter 6: Through the investigation of existing morphological texture segmentation techniques, an algorithm that utilises area morphology local granulometries has been developed [41]. Texture features derived from area morphology operators have the property of not introducing artificial artefacts to the feature space, thus, reducing the number of mis-classifications. By defining a feature space that comprises of connected image components at various scales the need for a decomposition consisting of multiple structuring elements is eliminated, removing the stage of calculating pattern spectrum moments.

Area morphology derived texture features provide a means for reducing the speckle noise component with the granulometric shrinkage algorithm. The incorporation of speckle reduction in to the feature extraction process leads to a more robust segmentation.

Chapter 1

Introduction

IN recent years there has been increased interest in the use of radar for imaging of the Earth's surface. This is partly because microwave propagation is almost independent of meteorological effects, providing an all-time all-weather surveying tool. In addition, microwaves are sensitive to the geometric and dielectric properties of vegetation making radar an ideal candidate for the monitoring of land cover types.

Information in the form of an image is carried at single-pixel level. However this is not the only type of information contained in an image. Image segmentation is one of the most important steps leading to the analysis of processed image data–its main goal is to divide an image into parts that have a strong correlation with objects or areas of the real world contained in the image. The primary objective of radar image segmentation is separation of the image into regions with homogeneous properties. Different regions or areas are often characterised by different textures and, for these cases, segmentation can be achieved by separating the image into regions with different textures. Applications of texture segmentation of radar images include

- Mapping the Earth's surface
- Crop identification and monitoring [127] [132] [145]
- Sea ice detection and tracking [123] [113]

- Measuring deforestation [94] [116] [37]
- Detecting oil slicks [15] [17]

This chapter introduces the topic texture segmentation in synthetic aperture radar (SAR) images. Section 1.1 describes how an image is formed by a SAR, while Section 1.2 is concerned with properties of coherent imaging systems. These properties highlight considerations that need to be taken into consideration when processing images of this nature. Section 1.2.2 defines data models that can be used in the simulation SAR images. The subject of image texture analysis is covered in Section 1.3, which details how textural features can be extracted from images. In Section 1.3.1 a method for using textural features to classify regions or objects of an image is described. If prior knowledge of the textures present in an image is available the classification can be achieved by supervised texture analysis, Section 1.3.2. The use of clustering algorithms to achieve unsupervised classification is explained in Section 1.3.3.

1.1 Synthetic Aperture Radar

Synthetic aperture radar (SAR) is a development of the sideways-looking airborne radar (SLAR) which is used to survey large strips, or swathes, of terrain. The radar is carried on an air or spaceborne platform moving at uniform speed and altitude. The forward motion provides scanning in the along track (azimuth) direction. The radar beam is directed to one side, perpendicular to the flight track and down towards the surface. The beam is wide in the range direction and so intersects the surface in an oval with the long axis extended in the cross track (range) direction, illustrated in Figure 1.1.

The major limitation of a radio frequency based single fixed-antenna SLAR system is the spatial resolution, as this is determined by the wavelength and size of the aperture. The antenna length partially specifies the area through which it collects radar signals, so the length of the antenna can be regarded as its aperture. With a larger antenna more information can be obtained about a particular viewed object, leading to a better image (improved resolution) of the object being formed. As the resolution increases to an

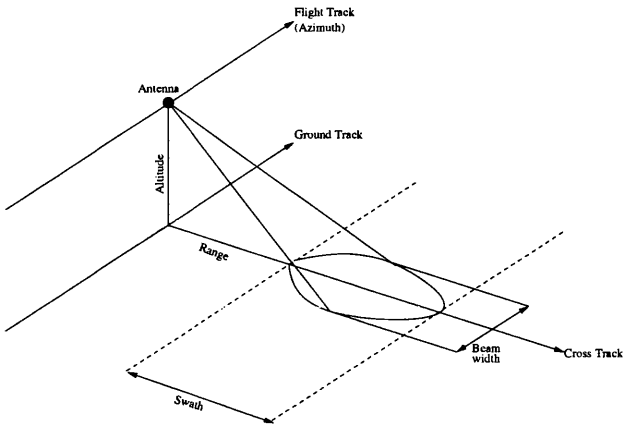


Figure 1.1: SAR Geometry.

appropriate value for imaging, the antenna quickly becomes too large to be carried by an aircraft or satellite. A way of obtaining higher resolution is to use the radar in SAR mode to overcome the limitations of technical or physical constraints on the aperture or antenna size.

With SAR it is possible to simulate a much larger antenna length by exploiting the forward motion of the radar platform by digital signal processing, giving the effect of a larger, synthetic aperture. If the radar is attached to a moving platform it is possible to combine reflected signals from along the flight path to synthesise a very long antenna. The synthetic aperture is the distance travelled by the platform whilst the target is illuminated by the beam. The width of the beam in azimuth is given by $2R\tan(\beta/2)$ where R is the range and β is the angle of divergence of the beam. For a SAR system, this corresponds to the maximum length of the synthetic aperture βR, as shown in Figure 1.2, with the three positions indicating:

1. Scatterer entering beam.
2. Scatterer broadside.
3. Scatterer leaving beam.

The forward motion of the platform causes a Doppler shift to the frequency of the returned echos. Consequentially the wavelength of echos from ahead of the centre line

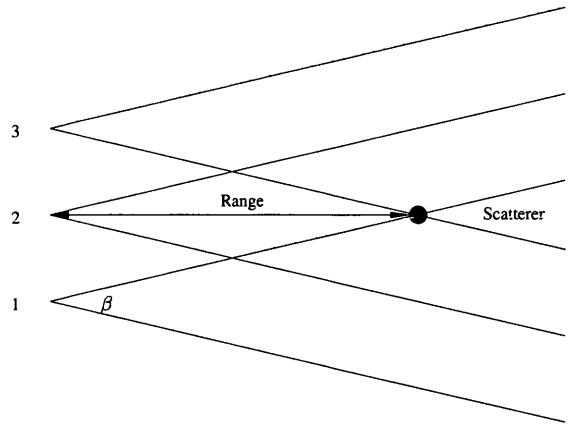


Figure 1.2: Plan view of SAR.

of view will be shortened and those from the rear will be lengthened. Therefore, if the receiver can discriminate frequency as well as time, and the speed of the platform is accurately known, it is possible to gather information about a particular target from many viewpoints as the platform moves forward, hence improving the resolution.

1.2 SAR Image Properties

Radar imaging systems are classed as coherent, defined by the systems ability to measure spatial and temporal variations in both the intensity of the scattered field and its phase. The quantity measured by a single frequency single polarisation SAR at each resolution cell is a pair of voltages in the in-phase and quadrature channels. These measured values represent the effects of the scene on the transmitted wave, and correspond to the geophysical quantity known as the radar cross section or RCS r. The observed in-phase and quadrature components, $z_i = A\cos\phi$ and $z_q = A\sin\phi$, can be represented by the complex number $A\exp(j\phi)$; in this form, the SAR data is known as the complex image.

The coherent nature of SAR images produces a noise-like characteristic known as speckle. Speckle is a universal property of coherent imagery and arises because the surface of the target is rough on the scale of the illuminating beam wavelength. A

rough target can be modelled as a collection of many randomly distributed point scatterers. As the wave interacts with the target, each scatterer contributes a backscatter wave with a phase and amplitude change. The observed intensity in a resolution cell can be regarded as resulting from the sum of contributions from each scatterer, and can be represented by,

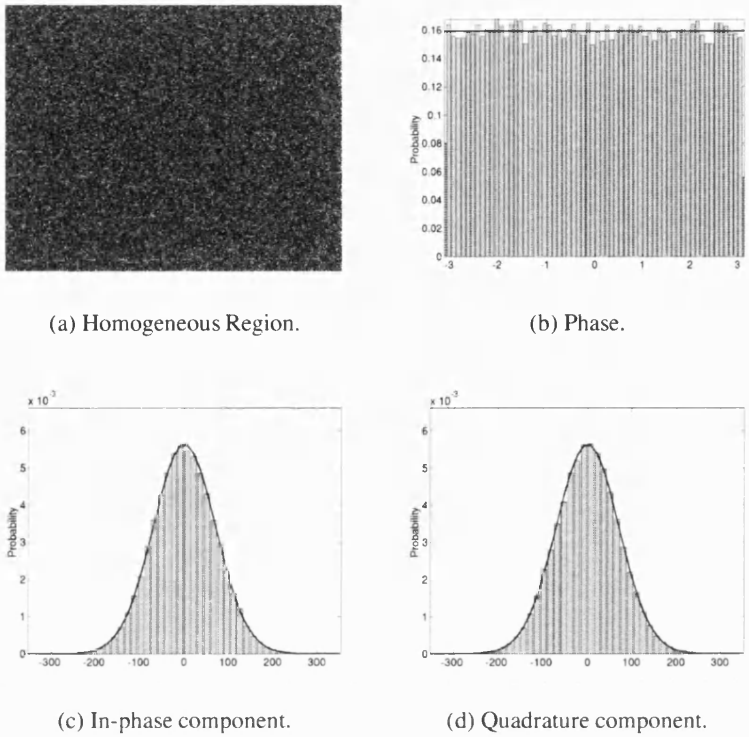
$$A \exp(j\phi) = \sum_{k=1}^{N} A_k \exp(j\phi_k). \tag{1.1}$$

This summation is over the number of scatterers, N, illuminated by the beam. The amplitudes A_k and phases ϕ_k are unobservable because the individual scatterers are on much smaller scale than the resolution of the SAR and there are many scatterers per resolution cell. Providing the number of discrete scatterers N is large, from the central limit theorem, it is expected that both the in-phase and quadrature components will be independent identically distributed Gaussian random variables with zero mean [93]. Using these models it is possible to simulate a homogeneous region in a SAR image. Figure 1.3 shows a simulated SAR image with an amplitude of 100 and compares its theoretical and simulated distributions.

As the slant range resolution of the SAR is typically many wavelengths across, scatterers at different parts of the resolution cell will contribute very different phases to the return signal, even if their scattering behaviour is identical. As a result the phases ϕ_k are uniformly distributed over the interval $-\pi \leq \phi_k \leq \pi$ and independent of the amplitude A_k, as shown in Figure 1.3(b). The summation (1.1) is now equivalent to a random walk in the complex plane, where each step of length A_k is in a completely random direction [52]. Figure 1.4 shows a random walk in the complex plane, illustrating how the phase and magnitudes of the independent scatterers make up the observed intensity in a resolution cell.

A conclusion from the random walk phenomenon is that the observed signal will be affected by interference as a consequence of the phase differences between scatterers. In fact, speckle can be understood as an interference phenomenon in which the principal source of the noise like quality of the observed data is the distribution of the phase terms ϕ_k.

At this point it must be stressed that speckle is noise-like, but is not noise; it is an



(a) Homogeneous Region.

(b) Phase.

(c) In-phase component.

(d) Quadrature component.

Figure 1.3: SAR image properties.

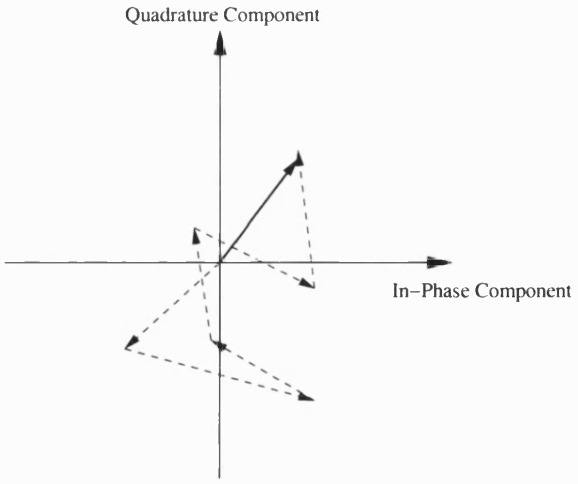


Figure 1.4: Random walk in the complex plane.

electromagnetic measurement. For SAR images the speckle noise is multiplicative and therefore dependent on the underlying radar reflectivity *i.e.* the noise level increases with the average intensity of a local region [73].

1.2.1 Product Model

It is important to distinguish the difference between the measured value at a pixel and the geophysical quantity measured by the SAR, the RCS r.

Figure 1.4 and (1.1) both indicate that the observed value at each pixel is the result of interfering backscatter waves from each discrete scatterer. Given that the value of r is specific to each pixel, the measured value is just a sample from the distribution parameterised by r. Assuming the amplitude and phase of each scatterer is independent of each other, and that the phases of different scatterers are independent and uniformly distributed, the mean intensity at a pixel is the incoherent sum of the mean intensity of all the scatterers, given by

$$\overline{I} = \overline{A^2} = \sum_{k=1}^{N} A_k^2. \tag{1.2}$$

Hence, the mean intensity $\overline{I}$ is independent of the geometrical configuration of the scatterers. However, changes in the positions of the scatterers will change all the phase relationships and result in altering the observed intensity, but without altering the parameter r.

The product model has been widely used in the modelling, processing and analysis of SAR images. It assumes that the observations within this kind of image are the outcome of a random variable that can be decomposed in the product of two other independent random variables: one modelling the speckle noise n, and the other modelling the underlying terrain backscatter r. Therefore, the observed intensity, I, in a SAR image can be expressed as the product

$$I = rn. \tag{1.3}$$

1.2.2 Speckle Model

Two assumptions are generally made about the contributions from the discrete scatterers within a resolution cell:

1. The amplitude A_k and phase ϕ_k of the k^{th} scatterer are statistically independent of each other and of the phases and amplitudes of all other scatterers.
2. The phases ϕ_k of each independent scatterer is uniformly distributed between the interval $-\pi \leq \phi_k \leq \pi$.

If these two assumptions are obeyed and the number of discrete scatterers k is large, the observed intensity obeys negative exponential statistics [52], *i.e.* the PDF is:

$$P_I(I) = \frac{1}{r}\exp\left(-\frac{I}{r}\right) \qquad I \geq 0, \tag{1.4}$$

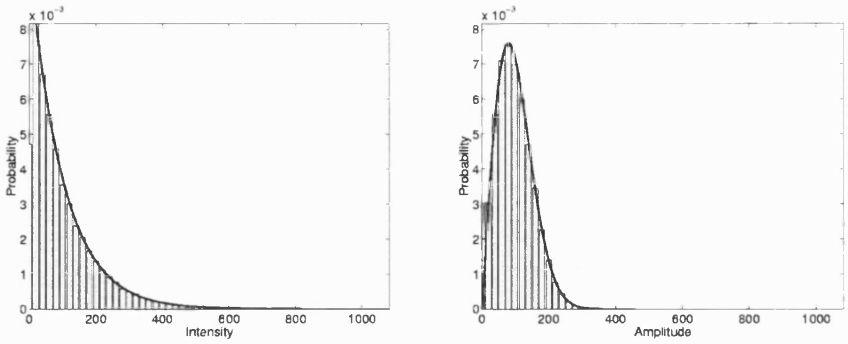
with mean value and standard deviation both equal to r.

The observed amplitude $A = \sqrt{I}$ will have a Rayleigh distribution:

$$P_A(A) = \frac{A}{r^2}\exp\left(-\frac{A^2}{2r^2}\right) \qquad A \geq 0, \tag{1.5}$$

with mean value $r\sqrt{\frac{\pi}{2}}$ and standard deviation $r\sqrt{\frac{4-\pi}{2}}$ [67]. Figure 1.5 compares the theoretical and simulated PDF's for 256 × 256 pixel amplitude and intensity images. The distributions (1.4) and (1.5) are completely characterised by a single parameter r, which carries all the information about the target.

One way to reduce the effects of speckle is to average together a number of images of the same scene, called looks [96]. This technique is know as multi looking and is covered in Section 3.1. Given L independent measurements of a scene, improved estimates of r at a given pixel can be gained by averaging the intensity. This is often called incoherent averaging because all phase information is discarded. The L look



(a) Negative Exponential Distribution.

(b) Rayleigh Distribution.

Figure 1.5: Theoretical and simulated speckle models.

average intensity is given by

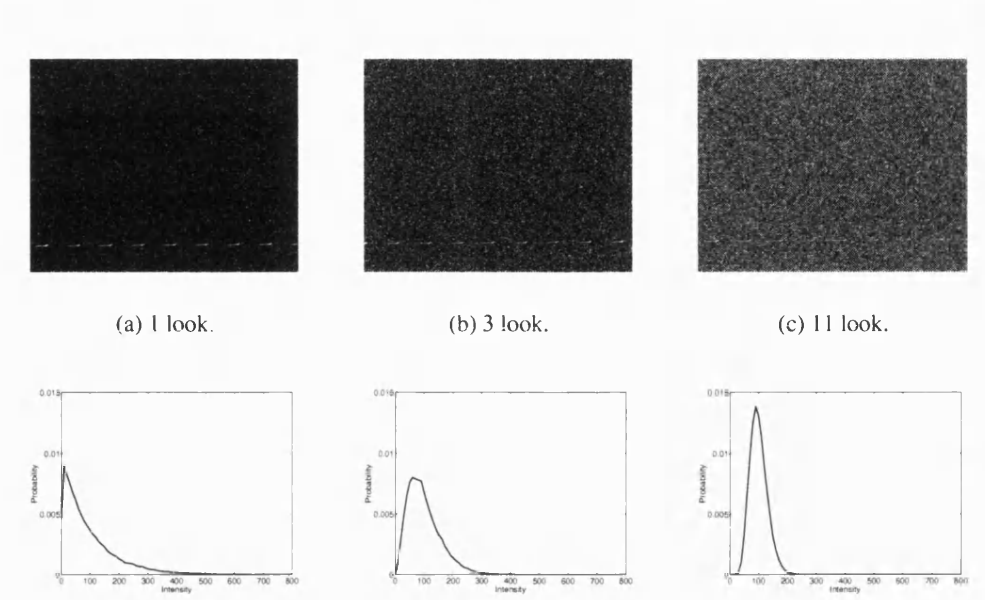
$$I = \frac{1}{L}\sum_{k=1}^{L} I_k, \tag{1.6}$$

where each of the I_k independent variables is exponentially distributed according to (1.4). The resulting distribution exhibits a gamma distribution [99], given by

$$P(I) = \frac{L}{r\Gamma(L)}\left(\frac{LI}{r}\right)^{L-1}\exp\left(-\frac{LI}{r}\right), \tag{1.7}$$

where L is the number of looks and Γ is the gamma function. Simulated images taken from the Gamma distribution are shown in Figure 1.6.

The other important aspect of speckle, which is sometimes neglected, is the second-order statistical characterisation. This is related, from a visual point of view, to the coarseness of its spatial structure or to the granularity of the image. Goodman [52] proposed this phenomenon, stating that a speckle pattern consists of peaks and nulls at many different scales. This granularity is attributed to an artefact of the imaging system that produces correlation between pixels [104] [115]. For a SAR "pixel" is not the same concept of quantity as "resolution". Resolution is fundamental and determined by the SAR bandwidth and processing strategy, whereas pixels are chosen to satisfy scale and sampling requirements. Normally there are two pixels per resolution cell in both the range and azimuth image co-ordinates. Also, the weighting functions used



(a) 1 look. (b) 3 look. (c) 11 look.

(d) 1 look pdf. (e) 3 look pdf. (f) 11 look pdf.

Figure 1.6: Multi look images of a homogeneous region.

in generating the data and in forming the looks generally introduce spatial correlation between pixels. For these reasons, realistic SAR image speckle simulation should include spatial correlation between pixels in two dimensions [104].

1.2.3 RCS Model

The product model described in Section 1.2.1 states that the measured intensity at a pixel in a SAR image is made up of two components, the geophysical property of the terrain (or radar cross section, RCS) and speckle noise. Once a model for the speckle noise component has been established, it is advantageous to have a knowledge of the underlying RCS distribution

Through studies of the statistical properties of sea clutter it was found that in regions of negligible swell the detected amplitude is Rayleigh-distributed, corresponding to pure speckle [140]. This implies that in homogeneous regions the distribution of a constant

is suitable to model the underlying RCS [44]. In regions of sea clutter where the swell component variations are on a much larger scale than the speckle component, the presence of speckle can be reduced by spatial averaging without degrading the swell component. Oliver and Queqan [96] have observed that the underlying swell RCS is usually consistent with a gamma distribution, given by

$$P(r) = \frac{r^{\alpha-1}}{\beta^{\alpha}\Gamma(\alpha)} \exp\left(-\frac{r}{\beta}\right), \tag{1.8}$$

where α and β are the shape and scale parameters of the Gamma distribution. It should be noted that (1.7) and (1.8) are equivalent with $\alpha = L$ and $\beta = r/L$. Frery *et al.* also state that a gamma distribution is suitable to model the terrain backscatter in heterogeneous (textured) regions [44]. The assumption of a gamma distribution (1.8) for modelling the terrain backscatter is mainly based on empirical evidence [43].

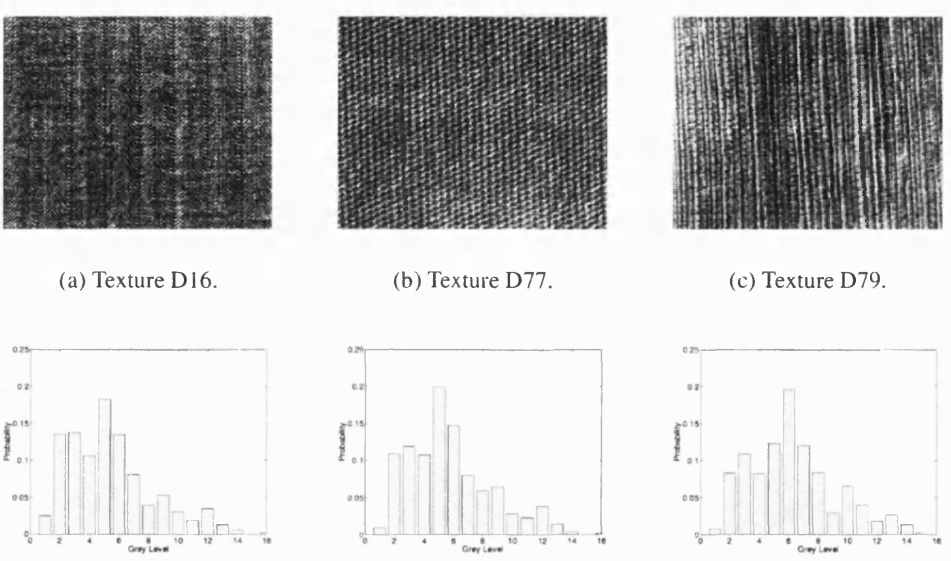
Correlated K distributions have been used to simulate clutter recorded by a SAR imaging process [18]. To simulate correlated K-distributions it is sufficient to apply negative exponential speckle to a correlated gamma distribution. A two-dimensional correlated gamma distribution can be approximated by the filtering of a gamma distributed white noise model with an appropriate filter to obtain the desired autocorrelation function.

1.3 Texture Analysis

Assigning a label to a textured region or object of an image is known as texture classification. This can only be achieved if knowledge of the spatial positioning of the regions or objects is known *a priori*. Assuming that pixels with similar textural features are located near each other in an image, texture segmentation can be achieved by performing texture classification on a pixel-wise basis. Textured regions of an image have constant texture if a set of local statistics, or other local properties, are constant, slowly varying or approximately periodic [121]. Texture analysis results in a summary or model of a textured region. The simplest form of summary is merely a list of texture features in a vector format. A texture feature vector can consist of values obtained by a statistical approach which has three categories: a) the use of power spectrum and autocorrelation function; b) the use of grey-level statistics, *e.g.* histograms

and co-occurrence matrices [56]; c) the use of local feature statistics, *e.g.* the statistical distribution of corners.

Histograms of a population define the statistical behaviour of the intensity at one position, and hence, are single point statistics. However textures that are visually different can exhibit similar first order statistics. Figure 1.7 gives an example of three such textures, taken from Brodatz's album [22] and quantised to 16 greylevels.



(a) Texture D16. (b) Texture D77. (c) Texture D79.

(d) Texture D16 pdf. (e) Texture D77 pdf. (f) Texture D79 pdf.

Figure 1.7: Brodatz textures and probability density functions

Texture information in an image is contained in the overall spatial relationship that the intensity values have to one another. That is, in a textured image the irradiances at different positions are statistically related. Statistical relations between a number of pixels in an image are defined by higher order statistics. The statistical behaviour of two pixels in an image can be described by the joint probability density function, which fully defines the second order statistics. Often the second order statistics are adequate to characterise a texture, therefore description of a textured region involves the estimation of the joint probability density function [135]. Nearest-neighbour grey-tone spatial-dependence matrices or grey level co-occurrence matrices (GLCM) provide a

measure from which all texture features are derived and were first proposed by Haralick *et al.* [56]. The intensity at each pixel of the textured image for analysis can be represented by a grey level. The grey level appearing in each pixel is quantised into N_g levels, with $G = \{0, ..., N_g - 1\}$ as the set of N_g quantised grey levels. A pixel (excluding those in the periphery of an image) has eight nearest neighbour pixels, see Figure 1.8.

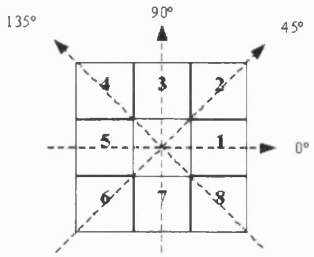
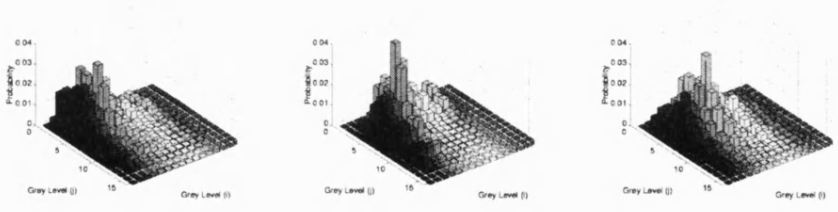


Figure 1.8: Nearest neighbour pixels.

A GLCM is a matrix of relative frequencies, $P(i, j)$, with which two neighbouring pixels, separated by a distance d and angle θ, occur on an image, one with grey level i, the other with grey level j. The resulting matrix is an N_gxN_g matrix with an entry at (i, j) for every neighbouring pixel pair. Note that GLCM are symmetric: $P(i, j; d, \theta)$=$P(j, i; d, \theta)$. After the number of neighbouring pixel pairs R, used in computing a particular GLCM is obtained, the matrix is normalised by dividing each entry in the matrix by R.



(a) Texture D16. (b) Texture D77. (c) Texture D79.

Figure 1.9: GLCM of Brodatz textures, d=10, $\theta = 45°$.

Figure 1.9 shows the GLCM for the Brodatz textures of Figure 1.7. Compared to the probability density functions, it can be seen that the GLCM provide better discrimination between the different textures. This reiterates the observatoin that the second order statistics are generally sufficient for characterising textures. A number of features can be derived from the GLCM to describe textures [76] [123], some of which are presented in Table 1.1, where the mean and standard deviations for the rows and columns of the matrix are:

$$\mu_x = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} iP(i,j) \qquad \mu_y = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} jP(i,j)$$

$$\sigma_x = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} (i-\mu_x)^2 P(i,j) \quad \sigma_y = \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} (j-\mu_y)^2 P(i,j)$$

Features derived from GLCM are common for the analysis of texture in SAR images [132] [123] [113] [111] [14].

1.3.1 Feature Based Texture Classification

The first stage of texture classification involves the development of a feature space that is used to describe the textures in an image. This process consists of calculating a number of features for each pixel (or group of pixels) in the image. Each individual feature should contain information about the appearance of the local neighbourhood surrounding the pixel. The set of individual features makes up a feature vector for each pixel, and allows each pixel to be represented as a point in a multidimensional feature space [100]. If the features used are good descriptors of the pixels surroundings, visually similar regions in the image will contain pixels whose feature vectors occupy similar positions in the feature space. These regions of similar feature vectors are known as clusters and identify a given texture class in the image. Image segmentation can be achieved by identifying these regions and assigning each feature vector to one of the texture classes.

An example of how a feature space can be defined when working with multi-spectral SAR data is to plot the response of each pixel in a multi-spectral vector space, with as

Energy	$\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} P(i,j)^2$
Contrast	$\sum_{n=0}^{N_g-1} n^2 \left\{ \sum_{i=1}^{N_g}\sum_{j=1}^{N_g} P(i,j) \right\}\Big\vert_{\vert i-j\vert=n}$
Correlation	$\dfrac{\sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(ij)P(i,j) - \mu_x\mu_y}{\sigma_x\sigma_y}$
Homogeneity	$\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} \dfrac{1}{1+(i-j)^2} P(i,j)$
Entropy	$-\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} P(i,j)\ln\{P(i,j)\}$
Autocorrelation	$\sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(ij)P(i,j)$
Dissimilarity	$\sum_{i=1}^{N_g}\sum_{j=1}^{N_g} \vert i-j\vert P(i,j)$
Cluster Shade	$\sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(i+j-\mu_x-\mu_y)^3 P(i,j)$
Cluster Prominence	$\sum_{i=1}^{N_g}\sum_{j=1}^{N_g}(i+j-\mu_x-\mu_y)^4 P(i,j)$
Maximum Probability	$\max_{i,j}\{P(i,j)\}$

Table 1.1: Texture features from grey level co-occurrence matrices.

many dimensions as there are spectral components [105]. In this space, each pixel in an image plots as a point with the co-ordinates given by the brightness value of the pixel in each spectral band. Provided there is good discrimination between spectral bands, it is expected the pixels will form clusters in the multi-spectral space corresponding to different ground cover types. Figure 1.10 illustrates an example of a two-dimensional feature space, showing how the responses from image pixels will plot in the feature space. The clusters of pixel points are referred to as information classes, since they are ground cover types used in the classification scheme. In practice the information classes may be portrayed as a group of clusters for the same ground cover type.

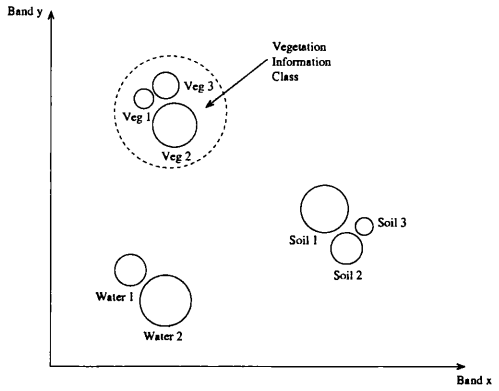


Figure 1.10: Classes in multi-spectral feature space.

Having defined the feature vector for each pixel, $\overline{x}$, it is now important to define the codevector μ_k for texture k. The positions of pixel points in the multi-spectral space can be described by vectors whose components are the individual spectral responses in each frequency band. The mean vector is useful to define the average or expected position for texture k in the multi-spectral vector space. For an image of sample texture k, the codevector is calculated by

$$\mu_k = \mathcal{E}\{\overline{x}\} = \frac{1}{N}\sum_{n=1}^{N}\overline{x}_n, \tag{1.9}$$

where $\overline{x}_n$ are the individual pixel vectors of total number N.

Although the mean vector μ_k defines the expected position of texture k in a multi-spectral space, it is of value to have available a measure of the features variance in

each frequency band, which can be observed from the covariance matrix

$$C_k = \frac{1}{N-1}\sum_{n=1}^{N}(\overline{x}_n - \mu_k)(\overline{x}_n - \mu_k)^T. \tag{1.10}$$

The covariance matrix also provides a measure of correlation between a pair of spectral bands, whereby a large value in the corresponding off-diagonal element of the covariance matrix indicates high correlation. Alternatively, a value close to zero implies little correlation between the spectral bands.

1.3.2 Supervised Texture Analysis

A set of local statistics can be represented as an N dimensional vector in feature space, providing a method for classification of image textures based on vector quantisation (VQ) techniques. VQ is a process for approximating a set of input vectors into a finite set $\Omega = \{\mu_k, 1 \leq k < K\}$. Ω is called a codebook and each element $\mu_k \in \Omega$ is called a codevector. With more features in each codevector the codebook can contain a greater number of discrete textures [102]. The classification of the feature vectors can be performed by various algorithms *i.e.* maximum likelihood classification, minimum distance classification or table look up classification [105]. In a minimum distance classification scheme the distance from the input vector to each codevector, μ_k, is calculated. The codevector with the minimum distance is assigned to the pixel represented by the input feature vector. Examples of distance functions used for the discrimination of feature vectors are given in Table 1.2. For a covariance matrix C_k, the variance of feature j for texture k is $c_k(j)$, represented by the diagonal of C_k.

For supervised classification the distance between each pixel feature vector and each texture in the codebook Ω is calculated by

$$d(\overline{x}, \mu_k) = \text{distance}(\overline{x}, \mu_k), \tag{1.11}$$

and the pixel is assigned to texture k if $d(\overline{x}, \mu_k) < d(\overline{x}, \mu_l)$ for all $k \neq l$.

Euclidean Distance	$d_1(\overline{x}, \mu_k) = \sum_{j=1}^{J} \{\overline{x}(j) - \mu_k(j)\}^2$
Bayes Distance	$d_2(\overline{x}, \mu_k) = (\overline{x} - \mu_k)^T C_k^{-1} (\overline{x} - \mu_k) + \ln \lvert C_k \rvert$
Mahalanobis Distance	$d_3(\overline{x}, \mu_k) = (\overline{x} - \mu_k)^T C_k^{-1} (\overline{x} - \mu_k)$
Simplified Mahalanobis Distance	$d_4(\overline{x}, \mu_k) = \sum_{j=1}^{J} \frac{\{\overline{x}(j) - \mu_k(j)\}^2}{c_k(j)}$

Table 1.2: Distance functions for classification, where $\overline{x} = \{\overline{x}(1), ..., \overline{x}(J)\}$ is the input feature vector, $\mu_k = \{\mu_k(1), ..., \mu_k(J)\}$ is the code vector and C_k is the covariance matrix of the feature set for texture k.

1.3.3 Unsupervised Texture Analysis

For texture segmentation it is necessary to assign every pixel of an image to a texture class [144]. This requires the calculation of a feature vector for every image pixel. Provided that the texture features are capable of discriminating the textures in the image, the feature vectors belonging to each texture class will form clusters in the feature space. If each cluster is compact and isolated from other texture classes, clustering algorithms can be used to recover the clusters in the feature space [61].

It was reported by Kohonen that the Self-Organizing Map has been successful at pattern recognition tasks involving a signal corrupted by noise, for example the radar classification of sea ice [66]. The Self-Organizing Map is an artificial neural network with cells specifically tuned to patterns in the input signal, and has the property of creating spatially organised representations of signal features and their abstractions. Each cell, or local cell cluster, acts as a separate decoder for each feature vector input. It is therefore the presence or absence, as opposed to the actual magnitude of the feature responses, that provides an interpretation of the input vector.

Provided the number of clusters, K, is know *a priori* the objective of cluster analysis is to divide a given input data set onto subsets, or clusters. The clustering process then has to verify the two constraints: 1) There are no empty clusters; 2) Every entry in the input data set has been classified [1].

1.3.3.1 K-Means Clustering

Vector quantisation, VQ, produces an approximation to a continuous probability density function of the vector input $\overline{x}$ using a finite number of codevectors $\Omega = \{\mu_k, 1 \leq k < K\}$. Once the codebook, Ω, has been defined, the approximation of $\overline{x}$ involves finding the codevector μ_c closest to $\overline{x}$. One method of positioning Ω in the feature space minimises ϵ, the expected r^{th} power error by

$$\epsilon = \int \|\overline{x} - \mu c\|^r p(\overline{x}) d\overline{x}, \tag{1.12}$$

where $d\overline{x}$ is the volume differential in the $\overline{x}$ feature space, and the index $c = c(\overline{x})$ of the best matching codevector (or *winner*) is a function of the input feature vector $\overline{x}$ such that

$$\|\overline{x} - \mu c\| = \min_k \{\|\overline{x} - \mu_k\|\}. \tag{1.13}$$

Generally no single solution for the optimal placement of Ω in the feature space is possible and an iterative scheme needs to be used. The optimal placement of Ω in the feature space by (1.12) results in the point density function being approximation to $p(\overline{x})^{\frac{n}{(n+r)}}$, where n is the dimensionality of $\overline{x}$ and μk [49]. In most applications $n \gg r$, and the optimal VQ will approximate $p(\overline{x})$.

K-means algorithm [66]:

If $\mu_c = \mu_c(t)$ is the Euclidean closest codevector to $\overline{x} = \overline{x}(t)$ (see Table 1.2) at a given time t, the steepest descent gradient-step optimisation of ϵ in the μ_c space produces the sequence

$$\begin{aligned} \mu_c(t+1) &= \mu_c(t) + \alpha(t)[\overline{x}(t) - \mu_c(t)] \\ \mu_k(t+1) &= \mu_k(t) \qquad \text{for} \quad k \neq c, \end{aligned} \tag{1.14}$$

where $\alpha(t)$ is a monotonically decreasing sequence of gain coefficients, $0 < \alpha(t) < 1$. If the dissimilarity between $\overline{x}$ and μ_k is expressed in terms of a general distance function (1.11), the *winner*, μ_c is identified by

$$d(\overline{x}, \mu_c) = \min_k \{d(\overline{x}, \mu_k)\}. \tag{1.15}$$

After the classification, an updating rule should be used such that the distance reduces monotonically. The correction $\delta\mu_k$ of μ_k is such that

$$[\text{grad}_{\mu_k} d(\overline{x}, \mu_k)]^T \cdot \delta\mu_k < 0. \tag{1.16}$$

For the updating operation the new codebook $\Omega(t+1)$ is identified by calculating the average of those $\overline{x}$ that are identified as belonging to class k.

1.3.3.2 Fuzzy C-Means Clustering

K-Means clustering is known as hard classification where an input feature vector can belong to only one cluster. Alternatively fuzzy-c clustering is based on the similarity (or membership) between feature vectors and clusters. The membership value, U, calculated with the fuzzy c-means algorithm, where $U = \{U \in \mathbb{R} \text{ and } 0 \leq U \leq 1\}$, is associated with the probability of an input set belonging to a given class. The fuzzy c-means clustering algorithm has been used in the clustering of wavelet features for texture segmentation in [23] [77].

Fuzzy c-means algorithm [23]:

Initially the number of clusters, $K = \{2 \leq K < N\}$ and the fuzzy weighting exponent, $m = \{1 \leq m < \infty\}$ (which controls the fuzziness of the clusters [15]), are set. Initialise the fuzzy membership function, $\overline{U}^{(0)}$, with random values and perform for iteration $t = \{t \in \mathbb{N}^+\}$.

1. Calculate fuzzy class centres, $\overline{v}_k$, with:

$$\overline{v}_k = \frac{\sum_{n=1}^{N} (u_{k,n})^m \overline{x}_n}{\sum_{n=1}^{N} (u_{k,n})^m}, \qquad k = 1, 2, \ldots, K, \tag{1.17}$$

   where $\overline{x}_n, n = 1, \ldots, N$ are the input feature vectors.

2. Update $\overline{U}^t$ with:

$$u_{k.n} = \frac{1}{\sum_{i=1}^{K} \left(\frac{d_{k,n}}{d_{i,n}}\right)^{\frac{2}{m-1}}}, \tag{1.18}$$

where $(d_{k,n})^2 = \|\overline{x}_n - \overline{v}_k\|^2$ and $\|.\|$ is the inner product norm.

3. Compare $\overline{U}^{(t)}$ with $\overline{U}^{(t-1)}$.

If $\|\overline{U}^{(t)} - \overline{U}^{(t-1)}\| \leq \epsilon$ stop; otherwise, return to step 1.

The outcome of the fuzzy c-means algorithm is the fuzzy membership function $\overline{U}^{(t)}$, which is a vector of dimension K. For a particular pixel in an image, $\overline{U}^{(t)}$ describes the membership value with respect to each cluster, and the sum of the elements is equal to 1. Once the final fuzzy membership function is obtained, a clustering map is produced by assigning each pixel to the class which it most probably belongs to.

1.4 Summary

This chapter introduced the basic properties of coherent imaging systems and how models can be used to simulate SAR images. These models are important for the topic of speckle noise removal covered in Chapter 3. Also introduced was the subject of image texture analysis. Texture feature extraction and a standard technique to achieve this, the grey level co-occurrence matrix (GLCM), was introduced. Once textural features have been formed the process of classification can be achieved, via supervised or unsupervised classification. Suitable clustering algorithms for unsupervised classification *i.e.* K-means and fuzzy C-means were described.

Chapter 2

Multi-Scale Image Analysis

THROUGH research into human and mammalian vision it was discovered that multiscale analysis provides the simultaneous localisation of energy in both the spatial and spatial-frequency domains [114]. This discovery has focused the research into texture analysis on the fields of multi-channel and multi-resolutional image analysis. This chapter discusses research of multiscale texture analysis, starting in Section 2.1 with an early texture perception model of the human visual system. The concept of Gabor filters is introduced in Section 2.2, with Section 2.2.1 describing the theory behind joint representation in the spatial and spatial-frequency domains. Section 2.3 describes how a bank of Gabor filters, with varying spatial bandwidth and orientation selectivity, can be constructed. The various filters are visualised in both the spatial and spatial-frequency domains to provide a description of the spectral content extracted by each filter in the filterbank. A modern tool for the multi-resolutional analysis of images, the wavelet transform, is introduced in Section 2.4. To enable the wavelet transform to be used for digital image analysis its discrete form must be employed. The discrete wavelet transform is described in Section 2.4.1 with its implementation via a two-channel filter bank is detailed in Section 2.4.2. The discrete wavelet transform can be utilised for texture analysis by detecting subbands of an image which contain textural information, this is achieved via the wavelet packet transform in Section 2.4.4. The concept of mathematical morphology is introduced in Section 2.5. Mathematical morphology can be used for the multi-scale analysis of images by the use of granulometries, as described in Section 2.5.1. Although mathematical morphol-

ogy is the analysis of images based on shape, Section 2.5.2 describes area morphology which is adaptive to local shapes of an image. Image segmentation can be achieved with the morphology based watershed transform detailed in Section 2.5.3.

2.1 The Human Visual System (HVS)

Texture analysis is believed to play an important role in the human visual system for recognition and interpretation. This insight lead to research in the area of human texture perception, with the the conclusion the human visual system is performing simultaneous analysis in both the spatial and spatial-frequency domains.

Silverman [114] presented research on the striate cortex[1] in primates. The aim of the work was to measure the spatial-frequency tuning of cells at regular intervals through the monkey striate cortex, to see if there is a periodic arrangement of cells tuned to different spatial-frequency ranges. Within an area that processes the visual input from a retinal region, cells were found to be tuned to all the different orientations in a systematic order. Along with orientation selectivity, it was found there was a periodic arrangement of cells tuned to different spatial-frequency ranges. The tuning of the cells changed systematically, varying from a low frequency range, through the midbands, to high frequencies and back again. Most cells in the striate cortex have both orientation and spatial-frequency tuning, and respond to a limited spatial-frequency range, thus in effect acting as a band pass two-dimensional filter within a localised region of the visual field.

2.1.1 Model for Texture Perception

The simulation of striate cortex cells with a bank of bandpass filters is extended in [78] to describe a model for human texture perception. The model presented consists of three stages:

[1]The region of the cerebral cortex of the brain where sensory information from the eyes is interpreted.

1. Convolution of the image with a bank of even-symmetric linear filters followed by half wave rectification, modelling the outputs of the striate cortex cells.

2. Localised inhibition among the cell response profiles, resulting in the suppression of weak responses when there are strong responses in the local neighbourhood.

3. Texture boundary detection using odd-symmetric mechanisms, resulting in edge detection of the texture map.

The point-spread function used in the model to simulate the responses of cells was the difference of Gaussians DOG. These filters were chosen because of their good fit with psychological measurements and their computational simplicity. The filters are designed by summing 2-dimensional Gaussian functions with a zero mean, defined as

$$g(x, y, \sigma_x, \sigma_y) = \frac{1}{2\pi\sigma_x\sigma_y} \exp\left\{-\frac{1}{2}\left[\left(\frac{x}{\sigma_x}\right)^2 + \left(\frac{y}{\sigma_y}\right)^2\right]\right\}. \tag{2.1}$$

The radially symmetric filters are the obtained by the summation of three Gaussians [78] given by

$$DOG(\sigma) = F(x, y) = a.g(x, y, \sigma_a, \sigma_a) + b.g(x, y, \sigma_b, \sigma_b) + c.g(x, y, \sigma_c, \sigma_c), \tag{2.2}$$

with the variance $\sigma_a^2 : \sigma_b^2 : \sigma_c^2$ in a ratio of $0.62 : 1 : 1.6$ and $a : b : c$ in a ratio of $1 : -2 : 1$. The summation (2.2) is illustrated in Figure 2.1, for simplicity the Gaussians are in 1-dimension, with $\sigma = 2$. For a full set of filters at a given spatial-frequency an additional orientation argument is required, $DOG(\sigma, \theta)$. Filters at the same spatial-frequency, paramaterised by σ_x and σ_y, but orientation tuned are obtained by rotation about the centre of $DOG(\sigma, 0)$.

The filtering process, which involves convolutions of the image with DOG filters, is linear. A model based purely on linear processes cannot reproduce the performance of the human visual system for texture segmentation [78]. If we consider two textures with identical mean intensity values, convolving them with a linear filter results in responses with identical spatial averages. Therefore a nonlinearity needs to be introduced into the process for texture perception. The input image $I(x, y)$ is convolved with a bank of linear filters F_k followed by half-wave rectification. The

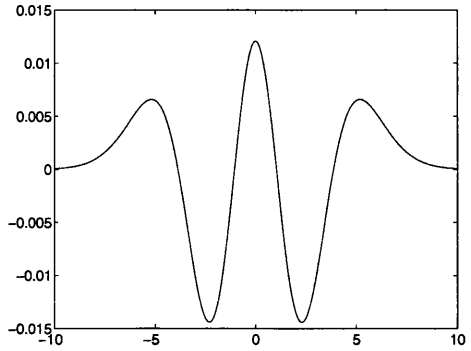


Figure 2.1: Summation of three Gaussians.

positive part is indicated by $R^+(x,y) = max(R(x,y),0)$ and the negative part by $R^-(x,y) = max(-R(x,y),0)$, which results in a set of neural responses $R_i(x,y)$, where i identifies the orientation-frequency channel. Therefore the convolution with each filter in the filterbank results in two response images,

$$R_{2k} = (I * F_k)^+(x,y), \qquad \text{and} \qquad R_{2k+1} = (I * F_k)^-(x,y). \tag{2.3}$$

There are two reasons for choosing half-wave rectification. Firstly linear filtering followed by half-wave rectification is a good first-order approximation for modelling responses of striate cortex cells [78]. Secondly with full-wave rectification,

$$R_k = |(I * F_k)(x,y)|, \tag{2.4}$$

the sign of the filter response is lost, leading to reduced texture discrimination in the response images. After the linear filtering and half-wave rectification stage, the final stage of the model is inhibition, localised in space, with the aim of suppression or reduction of spurious responses in non-optimally tuned channels. This stage is followed by edge detection of the texture map to predict the texture boundaries. The model presented was shown to discriminate between different texture pairs that matched well with experimental results with human observers. The inputs to the model were grey-scale images, which is the format of SAR intensity images.

2.2 Gabor Filters

An alternative point spread function to model cells in the striate cortex are Gabor functions [78]. Gabor functions were developed in the context of communication systems with the goal of providing local frequency analysis on signals. It has been shown that two dimensional realisations of Gabor functions approximate the receptive field of cells in the cat striate cortex [28].

Classically, images can either be viewed as a collection of pixels (in the spatial domain), or alternatively, as the sum of sinusoids over an infinite interval (spatial-frequency domain). Gabor proposed that spatial representation and spatial-frequency representation are opposite extremes of a continuum providing joint spatial and spatial frequency representations. Gabor's theory is applicable to texture analysis as perceptually different textures correspond to differences in local spatial-frequency content. Thus, texture segmentation can be achieved by decomposing an image in joint spatial/spatial-frequency representation and using the information to locate regions of similar spatial-frequency content.

2.2.1 Gabor Theory

Gabor proposed a new method of signal analysis [46], which provided both time and frequency analysis. Up until the time of the paper there were only two types of signal analysis:

1. Time analysis: based upon sharply defined instants of time.
2. Fourier analysis: based upon infinite wave-trains of precisely defined frequencies, where the relationship between time and frequency is given by

$$s(t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} S(\omega)\exp\{j\omega t\}\,d\omega \qquad S(\omega) = \int_{-\infty}^{\infty} s(t)\exp\{-j\omega t\}\,dt. \quad (2.5)$$

Once the spectrum of a signal has been obtained via Fourier analysis, detail is given about the range of frequencies in the signal at a given resolution. However, there is no information about the epoch. This is because the range of the Fourier integral is an infinite interval and is far from the everyday perception of signals. Fourier's theorem makes a description in time and description by the spectrum, two mutually exclusive methods. If the term *frequency* is used in the strictest of terms, which applies to infinite wave-trains, a *changing frequency* becomes a contradiction in terms; it is a statement involving both time and frequency. Humans respond strongly to changes when interpreting signals. These changes, or impulses, generate an infinite harmonic series in the frequency domain. However these features are transient — localised in time. Therefore when trying to simulate the performance of human signal analysis it is important to analyse time domain and frequency domain signal features simultaneously. For example a piece of music can have a definite time pattern as well as a frequency pattern. It is possible to leave the time pattern unchanged, and double the frequencies by playing the piece of music on a piano an octave higher. Conversely the same piece of music can be played in the same key, but at a different time. This shows both time and frequency analysis are complementary as opposed to mutually exclusive.

Gabor's alternative method of analysis represents signals in two dimensions, with time and frequency as co-ordinates. The two-dimensional representations of signals were called *information diagrams*, as areas in the diagrams are proportional to the number of independent data which they can convey. This is due to the fact that the frequency of a signal which is of a finite duration can only be defined with a certain accuracy, which is inversely proportional to the duration, and vice versa. An example of a signal represented in both the time and frequency domain is illustrated in Figure 2.2 by taking each feature as orthogonal co-ordinates. Figure 2.2 shows how an infinite sinusoid is represented by a vertical line, its frequency is exactly defined while its epoch is entirely undefined. Alternatively an impulse function has a sharply defined epoch, but its energy is uniformly distributed over the entire frequency spectrum, shown as a horizontal line. Next, Gabor illustrated how signals other than an infinite sinusoid or delta function can be represented in a time/frequency diagram. The example chosen was a sinusoid over a finite time period. To analyse the properties of a signal it is useful to examine physical effects of the signal on measurement instruments. The Fourier spectrum of a sinusoid over a finite time period is shown in Figure 2.3(a). A physical instrument used to record the frequency spectrum is a bank of reeds, or

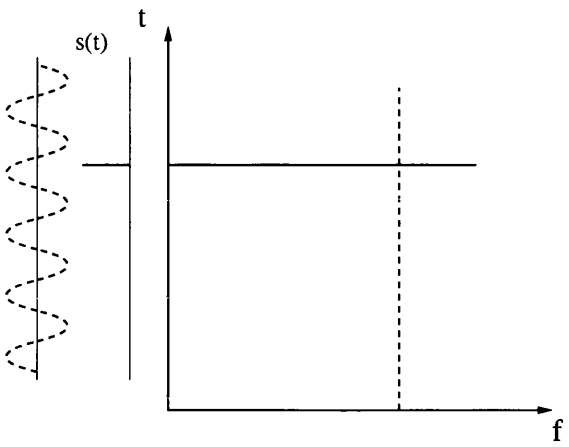
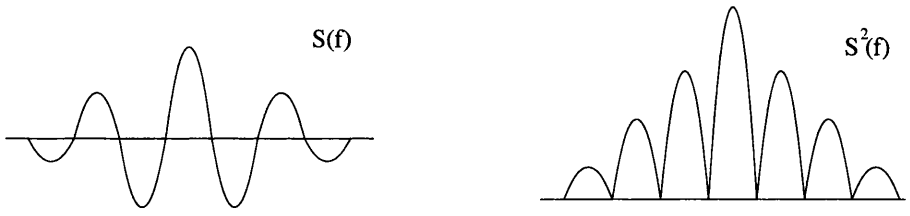


Figure 2.2: Infinite sine wave (dashed line) and impulse (solid line) in time/frequency domain.

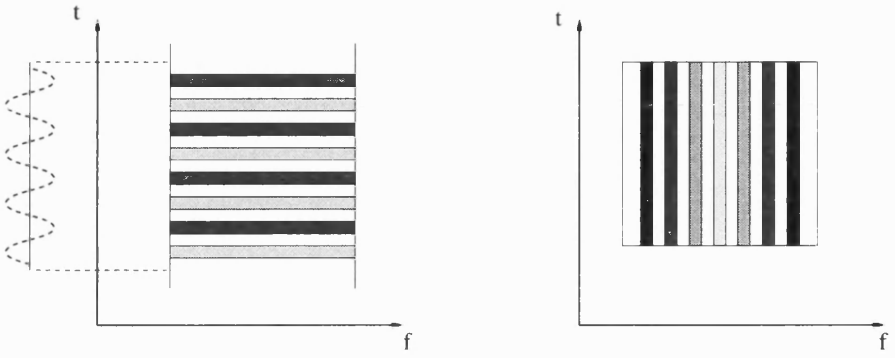


(a) Fourier spectrum.

(b) Energy spectrum.

Figure 2.3: Frequency responses of a finite sine wave.

other resonators, each tuned to a defined band of frequencies. Such an instrument only provides analysis of the energy spectrum, see Figure 2.3(b), but is adequate for this example of information diagrams.



(a) Oscilloscope. (b) Bank of reeds.

Figure 2.4: Time/frequency response of physical instruments to a finite sine wave.

The signal to the left of Figure 2.4(a) is the response of an ideal oscilloscope with uniform response over an infinite frequency range. Figure 2.4 compares the responses of an oscilloscope to a bank of reeds. For any instrument, resonator or oscilloscope, a damping time can be defined after which the oscillations have decayed by a given value *e.g.* 10dB. Similarly a tuning width can be defined which is the frequency of resonance at which the response falls by 10 dB. The relationship between the 2 parameters for a resonator is given by Equation (2.6). For every resonator a characteristic rectangle of constant area can be defined in a time/frequency diagram, which corresponds to a single independent reading from the instrument.

$$\text{Decay time} \times \text{Tuning width} = \text{Constant} \tag{2.6}$$

The division of the time/frequency area in constant rectangles is illustrated in Figure 2.4. For the oscilloscope the rectangles are broad horizontally and narrow vertically and vice versa for the bank of reeds. The number of rectangles in any time/frequency area is the number of independent data sample which an instrument can obtain from a signal, *i.e.* proportional to the amount of information, thus giving justification for calling such representations *information digrams*.

2.3 Gabor Filterbanks

A two dimensional Gabor filter is composed of a sinusoidal plane wave, modulated by a two dimensional Gaussian envelope (2.7). An important property of a Gabor filter is the optimal joint resolution in both the spatial and spatial-frequency domains. A two dimensional Gabor filter is given by

$$h(x,y) = \frac{1}{2\pi\sigma_x\sigma_y}\exp\left\{-\frac{1}{2}\left[\left(\frac{x}{\sigma_x}\right)^2+\left(\frac{y}{\sigma_y}\right)^2\right]\right\}\cos(2\pi f_0+\phi), \tag{2.7}$$

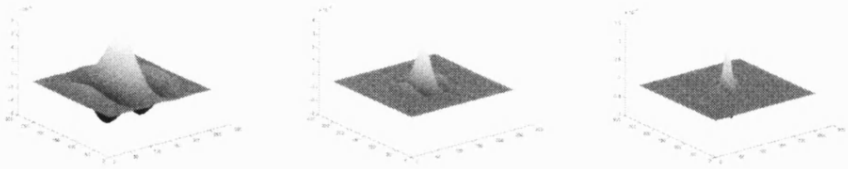
where f_0 and ϕ are the frequency and phase of the sinusoidal plane wave, and σ_x and σ_y are the space constants of the Gaussian envelope, which determine the Gabor filter bandwidth. The relationship between the radial frequency of the plane wave and space constants of the Gaussian envelope are

$$\sigma_x = \frac{N_c}{2f_0} \qquad \text{and} \qquad \sigma_y = \frac{N_r}{2f_0}, \tag{2.8}$$

where N_c and N_r are the number of rows and columns of the input image respectively. The Gabor filter radial frequencies were chosen to be one octave apart, as the cells in the human visual system have experimentally been shown to have the same frequency bandwidth [61]. For an image of width N_c the values of radial frequency f_0 used are

$$f_0 = 1\sqrt{2}, 2\sqrt{2}, 4\sqrt{2}, \ldots, (N_c/4)\sqrt{2} \qquad \text{cycles/image width}$$

The realisation of three Gabor filters is given in Figure 2.5, which shows the variation



(a) $f_0 = 2\sqrt{2}$. (b) $f_0 = 4\sqrt{2}$. (c) $f_0 = 8\sqrt{2}$.

Figure 2.5: Gabor filters of different spatial frequencies.

of the filter bandwidths with change in radial frequency. However, cells in the striate

cortex are also orientation tuned. A Gabor filter of arbitrary orientation, θ_0, can be obtained by rotation of the x-y coordinate system by $\theta°$ [61]. The Gabor filters in Figure 2.5 are all at an orientation of 0°, along the x-axis. The frequency response of a Gabor filterbank at all radial frequencies and 4 orientations is shown in Figure 2.6. The four values of orientation 0°, 45°, 90° and 135° are shown to provide non-overlapping responses in the frequency domain.

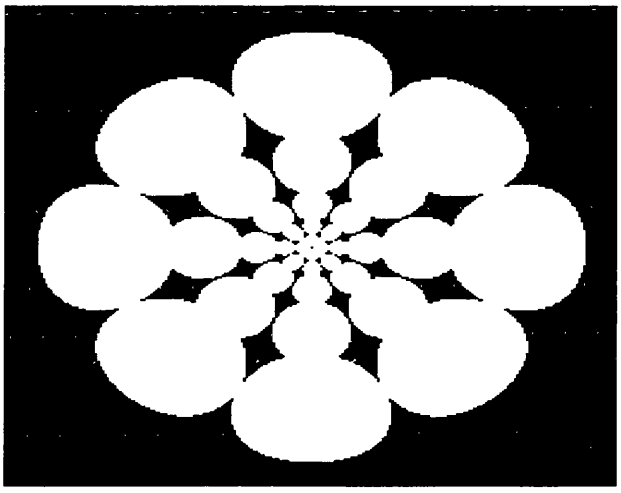


Figure 2.6: Gabor filterbank in the frequency domain.

2.4 Wavelet Transform

The *continuous wavelet transform* is a fairly modern signal analysis tool that has the advantage of representing time domain and frequency domain features simultaneously. This was originally proposed by Gabor [46] and is described in Section 2.2. Wavelets are not single frequency, but are localised waves, which means instead of infinite oscillations they drop to zero in a finite time.

The wavelet transform provides a method of *multiresolition* analysis, which is achieved by examining a signal over a range of scales. Wavelets are basis functions in the time domain. Each basis function, $\Psi_{jk}(t)$, is derived from the *mother wavelet*, $\Psi(t)$, through

a series of translations k, and changes in scale j [129]. An example of the Daubechies 12-tap basis function is shown in Figure 2.7.

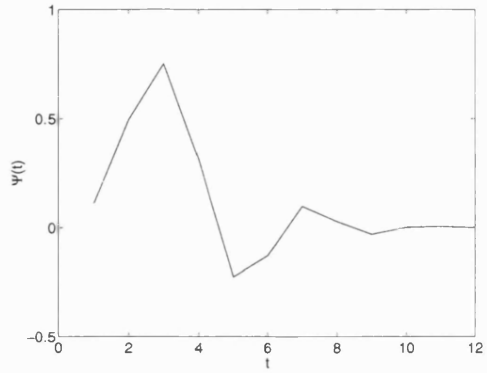


Figure 2.7: Daubechies 12 tap basis function.

Once $\Psi(t)$ has been defined the a family of other wavelets can be obtained by

$$\Psi_{jk}(t) = \frac{1}{\sqrt{j}}\Psi\left(\frac{t-k}{j}\right) \qquad \text{where } j > 0 \text{ and } k \in \mathbb{R}. \tag{2.9}$$

By convention the scale of $\Psi(t)$ is 1 and is centred around 0, therefore $\Psi_{jk}(t)$ has a change in scale equal to j and is centred around k. A signal of finite energy $x(t)$ can be represented as a linear combination of the family of wavelets $\Psi_{jk}(t)$, with the coefficients of the combination given by

$$c_{jk} = \int_{-\infty}^{\infty} x(t)\Psi_{jk}(t)dt. \tag{2.10}$$

The coefficients in (2.10) measure the fluctuations of the signal $x(t)$ at the scale j around the point k [86].

A measure of the wavelet transform's time analysis is given by the parameter Δt – the minimum time interval that can be resolved. Similarly, the frequency resolution is given by the parameter Δf. An increase in scale improves the frequency resolution whilst decreasing the time resolution. This implies the product of the time interval Δt, and the frequency interval Δf, is a constant. In fact the Heisenberg uncertainty

condition

$$\Delta t \Delta f \geq \frac{1}{2}, \tag{2.11}$$

gives a lower limit for the product. This equation represents the trade-off between accurate knowledge of a signal in the time domain and accurate knowledge in the frequency domain.

2.4.1 Discrete Wavelet Transform (DWT)

In the area of image processing the continuous wavelet transform is replaced by the *discrete wavelet transform* (DWT). For the case of the DWT the mother wavelet is translated and dilated by discrete values. At different scales the frequency interval, Δf, goes up by 2^j and the time interval, Δt, goes down by 2^j. The translated and dilated basis functions derived from the mother wavelet for the DWT are given by:

$$\Psi_{jk}(n) = 2^{-j/2}\Psi(2^{-j}n - k) \tag{2.12}$$

The DWT of a one dimensional signal is comprised by decomposing the input signal into translated and dilated forms of the mother wavelet. The wavelet decomposition of a signal $x(n)$ is given in (2.13).

$$x(n) = \sum_j \sum_k c_{jk}\Psi_{jk}(n) \tag{2.13}$$

where c_{jk} are called wavelet coefficients given by (2.14).

$$c_{jk} = \sum_t x(n)\Psi_{jk}(n) \tag{2.14}$$

The resolution properties of the discrete wavelet transform are examined in Figure 2.8, which shows how both the time and frequency resolutions should be interpreted. The regions in Figure 2.8 correspond to a wavelet coefficient (2.14) in the time-frequency plane. Therefore all the points in the time-frequency plane that fall into a given region are represented by a single coefficient of the wavelet transform. The regions are of

a finite area which implies the exact position of a point in the time-frequency plane cannot be defined. Although the dimensions of the regions change, one thing to be noted from Figure 2.8 is that the area remains uniform. The dimensions of the lower frequency regions are short height, which implies good frequency resolution and long width, which implies poor time resolution since there is more ambiguity regarding the exact positioning of the time. Alternatively at higher frequencies the width of the boxes decrease, giving an improvement in time resolution, and the hight increases giving a poorer frequency resolution. The area of all the regions is determined by Heisenberg's inequality (2.11).

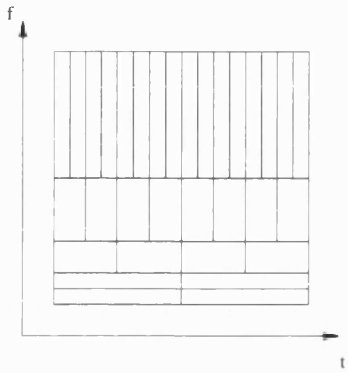


Figure 2.8: Resolution in time-frequency domain.

2.4.2 DWT Filterbank

Mallat [79] developed an algorithm for implementing the wavelet decomposition with a two-channel filterbank using quadrature mirror filters, see Figure 2.9. The filterbank is a set of filters linked by sampling operators [129]. The wavelet coefficients are acquired via a two-channel filter bank, one channel is low-pass and the other is high-pass. After the down-sampling operators the input signal, $x(t)$, has been decomposed into two half-resolution subbands. The inverse wavelet transform is implemented by the *synthesis* filterbank. A reconstruction of the input signal, $\hat{x}(t)$, is achieved by the summation of up-sampled and synthesis filtered channels.

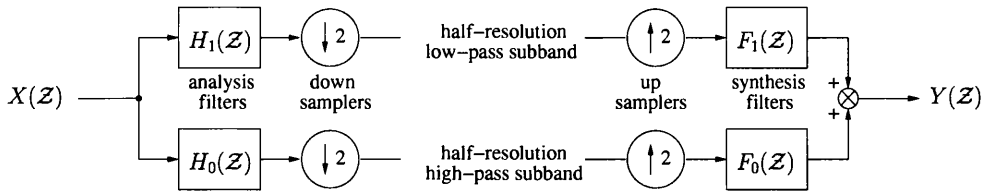


Figure 2.9: Two-channel wavelet filterbank.

2.4.3 Two Dimensional DWT

The DWT is a separable transform, thus an n dimension can be calculated by successively applying a one dimensional transform in each direction. For image analysis a two dimensional DWT can be implemented by applying a one-dimensional high and low pass filtering step to both the rows and columns to the input image. Each filtering step is followed by subsampling which results in a change in scale. If the same analysis filter is applied to the resultant signal, this is equivalent to changing the scale in the continuous wavelet transform. The filterbank to implement the wavelet transform is shown in Figure 2.10.

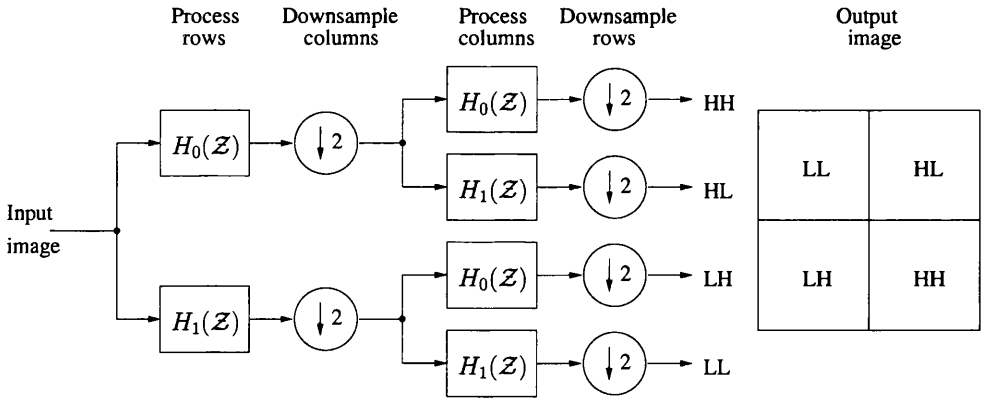


Figure 2.10: Discrete wavelet transform two-dimensional analysis filterbank.

At each decomposition level there are four different output images, an approximation of the input image and three detail images [50]. A description of the information contained in the output subbands of the wavelet transform are given in Table 2.1.

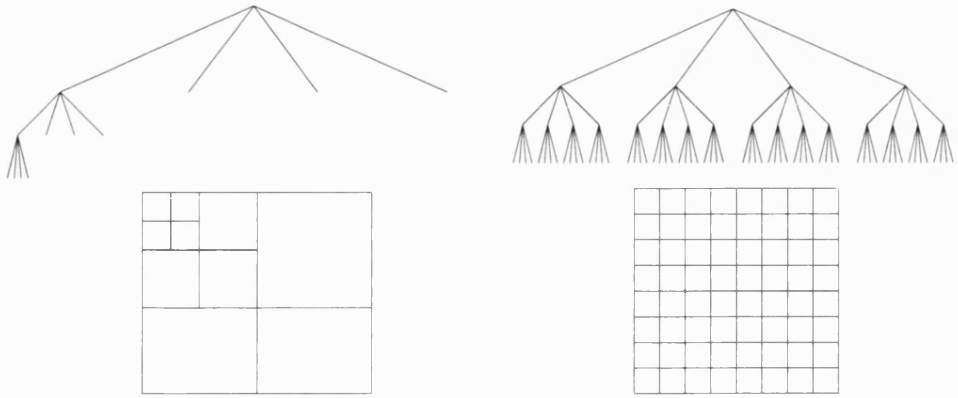
Subband	*Image detail*
LL	An approximation of the input image.
HL	Information on scale variations in the horizontal direction. A high value indicates the presence of a vertical edge.
LH	Information on scale variations in the vertical direction. A high value indicates the presence of a horizontal edge.
HH	Information on scale variations in the diagonal direction. A high value indicates the presence of a corner point.

Table 2.1: Information in wavelet subbands.

2.4.4 Wavelet Packet Transform

The conventional method of implementing the DWT in image processing applications is to recursively decompose the LL subband, known as the *pyramid* transform, see Figure 2.11(a). As the range frequency components are halved at each scale this transform is also known as the octave band decomposition [112]. Therefore the pyramid transform results in a logarithmic decomposition of the frequency axis, which provides good frequency selectivity and low frequencies, and good time selectivity at higher frequencies, see Figure 2.8. This property is well suited to signals which contain long-duration low-frequency components and short-duration high-frequency components. This makes the pyramid transform an ideal tool in applications geared towards image compression, where the majority of the information in an image is in the low-frequency region [3]. This matches the sensitivity of the human visual system, which decreases at high frequencies.

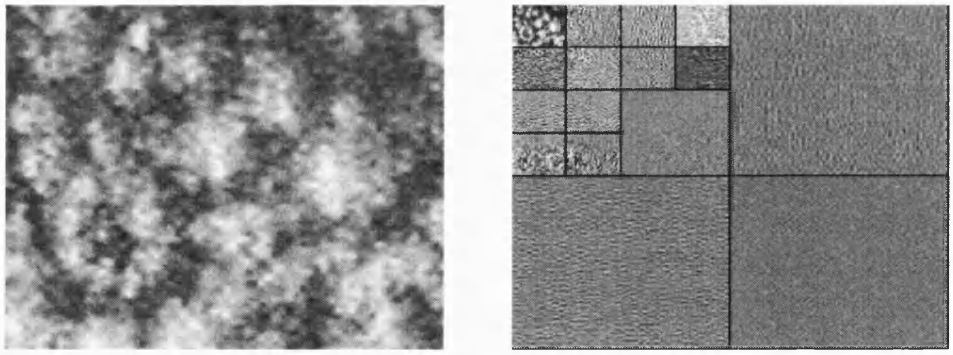
The pyramid transform is therefore inadequate for signals which contain mid-frequency stationary components. It has been stated that textural information is common in the mid-frequency bands of a signal [24]. To aid with textural feature extraction the wavelet transform has been adapted to the wavelet *packet* transform, where decomposition is performed all the output subbands listed in Table 2.1. As an alternative to a full wavelet packet decomposition, the tree-structured wavelet transform [24] (see Section 5.1.1), can be implemented to identify frequency channels which contain textural information. An example of a wavelet tree decomposition is shown in Figure 2.12, illustrating the dominant frequency channels of an input image.



(a) Wavelet pyramid transform.

(b) Wavelet packet transform.

Figure 2.11: Subbands of the wavelet pyramid and wavelet packet transforms.



(a) Sample SAR image consisting of an area of trees.

(b) Wavelet tree decomposition.

Figure 2.12: An example of the tree-structured wavelet transform, highlighting dominant frequency channels within an image.

The wavelet packet transform provides a mechanism for identifying a specific frequency range of a signal for analysis, and has been used for the texture analysis of digital images in [24] [23] [77] [69] [112].

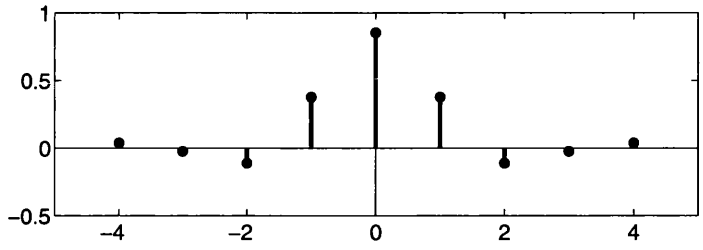
2.4.5 Redundant DWT

The lack of shift invariance has been acknowledged as on of the major drawbacks of orthonormal wavelet representations, with the result that a small shift in the space domain can result in a dramatic change in the wavelet domain [75]. The shift sensitivity is caused by the subsampling operators in each subband. Downsampling each subband by a factor of 2 means keeping the even numbered indexes and discarding the odd ones. If we consider the case where the input signal is shifted by one sample, the output signal is also shifted by one sample. Subsampling with the same operator results in keeping the odd numbered indexes and discarding the even ones. Thus, a completely different set of data points is produced [70]. By relaxing the requirement for critical sampling in the subbands, shift invariance can be established. A change in dilation in the basis function, $\Psi_{jk}(t)$, is then achieved by inserting an appropriate number of zeros between filter taps at each scale, see Figure 2.13.

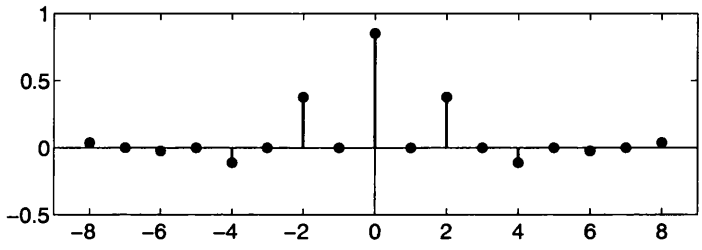
2.5 Mathematical Morphology

Mathematical morphology is a signal analysis tool based on spatial structures. Mathematical morphology was developed in the 1970's [83] by G. Matheron [82] and J. Serra [110]. It is called morphology because analysis is aimed on the shape and form of objects, and mathematical because the analysis is based on set theory and Boolean algebra. An introduction to set theory is given in Appendix C. Morphology has several advantages over convolutional filtering techniques, a few of which are:

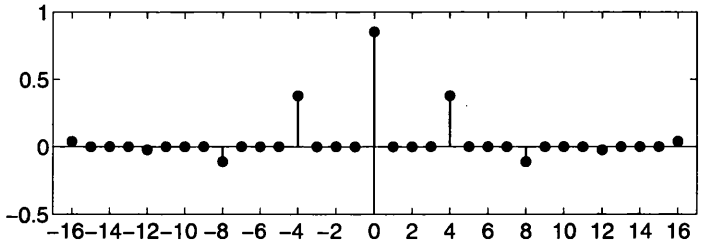
- The analysis is based on objects
- Preservation of edges



(a) Scale = 1.



(b) Scale = 2.



(c) Scale = 3.

Figure 2.13: Daubechies 9/7 low pass analysis filter taps.

- Non-linear operators
- Can produce idempotent filters

Mathematical morphology operators were originally defined to process binary images [82] [110] and subsequently extended to greyscale images, see Section D.2. The basic operators of mathematical morphology are *erosion* and *dilation*, defined by

$$X \ominus B = \bigcap_{x \in X} \{x - b | b \in B\}, \tag{2.15}$$

and

$$X \oplus B = \bigcup_{x \in X} \{x + b | b \in B\}, \tag{2.16}$$

respectively. The operations work with two functions [74]: the image to be analysed X and a structuring element B. An analogy of the structuring element is the kernel of a convolution operation. Dilation is the morphological operator that combines two sets using vector addition of each set element [57]. When applied to signal and images, dilation is an expansion operator of features. Erosion is the morphological dual of dilation, and is the operator that combines two sets using vector subtraction of set elements. Erosion has the effect of reducing features in signals and images.

A linear low pass filter smooths an image by removing high frequency components. An ideal lowpass filter will completely remove high frequencies whilst preserving all low frequency information. Therefore, if an ideal filter is applied to an already filtered signal, the signal will remain unchanged. A filter which possesses this property is known as *idempotent*. Idempotent operators are not possible with any practical realisation of a linear filter [13]. The prospect of designing an ideal filter which completely and irreversibly removes signal components is attractive. Morphological erosion and dilation operators discard signal information and can be combined to create idempotent filters.

Although erosion is an anti-extensive operator when applied to a signal, a subsequent dilation with the same structuring element will reverse some of the components removed from the signal. If the same operations are then applied for a second time, the resultant signal will be unchanged, hence the filter is idempotent. This sequence of

operations is known as a morphological *opening*

$$X \circ B = (X \ominus B) \oplus B, \tag{2.17}$$

which has the effect of eliminating sharp features in a signal smaller than the size of the structuring element used. The morphological dual of an opening is a *closing*

$$X \bullet B = (X \oplus B) \ominus B, \tag{2.18}$$

which is achieved by a dilation followed by an erosion. A morphological closing of an signal has the effect of fusing gaps smaller than the structuring element and eliminating small holes.

The result of iteratively applying morphological erosions and dilations to images eliminates image detail smaller than the structuring element, whilst preserving larger scale image features [57]. This means that by analysing an image with morphological filters it is possible to isolate features of interest, without adding new features at the corresponding scale.

2.5.1 Granulometries

The concept of granulometries is comparable to the sieving of materials of various sizes and can be implemented by morphological open-close operators. At each scale only the image structures too large to pass through the sieve remain. This method of decomposition, known as a datasieve [13], has the advantage of preserving the position and scale of sharp edged image structures and can be represented by a collection of image transforms,

$$\Phi = \phi_\lambda, \qquad \text{where } \lambda = \{1, 2, \ldots, \Lambda\}. \tag{2.19}$$

Datasieves using the morphological open-close operator

$$X \odot B_\lambda \qquad \text{where } \lambda \geq 1, \tag{2.20}$$

have been constructed in [13] [12] and have the following properties:

Increasing: $X \subseteq Y \Rightarrow \phi_\lambda(X) \subseteq \phi_\lambda(Y)$

When sieving a subset, the structures remaining after the sieve are a subset of the structures remaining after sieving the superset.

Anti-extensive: $S_\lambda \subseteq S_0$

The structures remaining after a sieve operation are a subset of the initial structures.

Absorbtion: $\phi_\lambda \phi_\mu = \phi_\mu \phi_\lambda = \phi_{\max(\lambda,\mu)}$

If an image is sieved with B_λ and B_μ, the order in which the sieves are applied is irrelevant as the result is determined by the sieve with the largest SE.

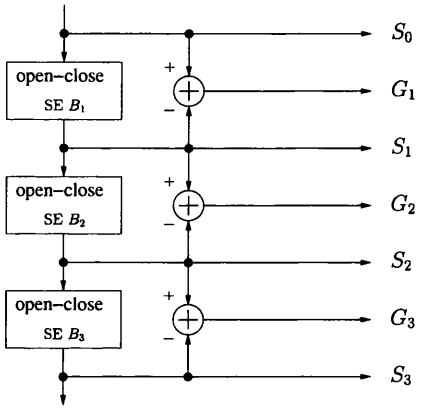


Figure 2.14: Granulometric sieve.

The structure of the datasieve in Figure 2.14, is a set of increasing scale open-close filters producing lowpass, S_λ, and bandpass, G_λ, outputs [13]. The datasieve yields an invertible image transform that express an image I as the summation

$$I = S_\lambda + \sum_{i=1}^{\lambda-1} G_i. \tag{2.21}$$

The residual between $S_{\lambda+1}$ and S_λ is a granule function, G_λ, which is composed of extrema of scale λ. The decomposition of an image into increasing scale granule functions represents the image information in a manner that is analogous to the wavelet pyramid transform [12].

The datasieve has been used for pattern recognition in [11], where at each stage the data is median filtered with window sizes $\{3 \times 3, 5 \times 5, \ldots, (2\Lambda+1) \times (2\Lambda+1)\}$. Textural features derived from the granulometric decomposition of an image can be used for image segmentation, this is achieved by single pixel classification [35] [34] [33].

2.5.2 Area Morphology

The basic morphological operators detailed in Appendix D can only describe image structures that have a shape-bias due to the SE. This shows a limitation of conventional morphological operators, since natural images can contain far more structures that can be described by a family of SE's, B_λ. In attribute morphology a family of SE's is defined with various properties. An attribute based approach to morphology provides a method of implementing many translation invariant operations, to remove connected components from an image based on a predefined criterion [20].

An operator that removes all structures from an image with an area smaller than a parameter α is called an *area opening*. When used with its dual *area closing*, this provides a method of removing the structures of an images whose area size is $\leq \alpha$. This form of attribute morphology is called *area morphology*. By defining a family of SE's to contain all shapes formed by α connected pixels, this describes a morphological operator without shape bias. When the family of SE's is used in a morphological opening this removes maximum extrema that are smaller than α, whilst preserving structures whose size is larger than α. Similarly, a morphological closing will remove minimum extrema from a signal. Area openings and closings have been applied to grey-scale images in [137] [138].

If B_α describes the family of SE's to contain all shapes formed by α connected pixels, a problem with this approach is the fast increase in the number of elements in B_α as α increases, see Table 2.2.

The rate of increase in B_α is sharper if the 8 nearest neighbours are included in the connected pixels. Figure 2.15 shows the family of connected components for an area size of 2 when considering the 8 nearest neighbours.

Area Size	Connected components
2	4
3	20
5	> 1000

Table 2.2: Connected components within 4 nearest neighbours.

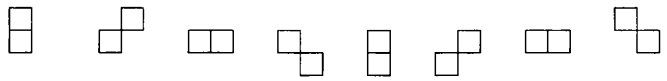


Figure 2.15: Connected components within 8 nearest neighbours with an area = 2.

Morphological area openings and closings [136, 107] remove connected components less than a given area and can be used to generate a scale space in which area provides a scale parameter equivalent to structuring element size in standard morphology. Area openings and closings can be defined by operations on a threshold decomposition or, alternatively, by

$$\gamma_\lambda^a(I) = \bigvee_{B \in A_\lambda} (I \circ B), \tag{2.22}$$

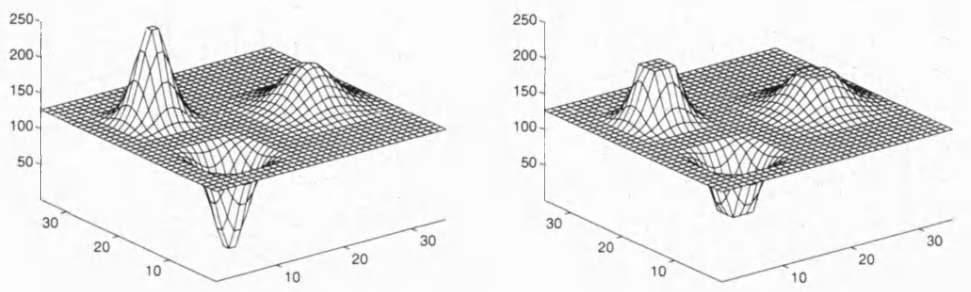
and

$$\varphi_\lambda^a(I) = \bigwedge_{B \in A_\lambda} (I \bullet B), \tag{2.23}$$

respectively where A_λ is the set of connected subsets with area greater or equal to λ [26]. This alternative definition illustrates the fact that area operators select the most appropriately shaped structuring at each pixel, the only constraint being its area. Therefore, the operations adapt to the underlying image structure and eliminate any shape bias and artificial patterns associated with fixed structuring elements.

Generating the set of SE's for large α, B_α, that contains all connected pixels of area size α is virtually an impossible task, which means there is no practical method to compute area morphology in this manner. This problem can be overcome by the use of area morphology, which is based on the observation that only local maxima (resp. minima) are affected by area openings (resp. closings). With the development of fast algorithms [85] it is now practical to implement area morphological operators in the field of digital image processing. An example of grey-scale area morphology is shown

in Figure 2.16, which shows how local extrema of a given area size are removed from an image.



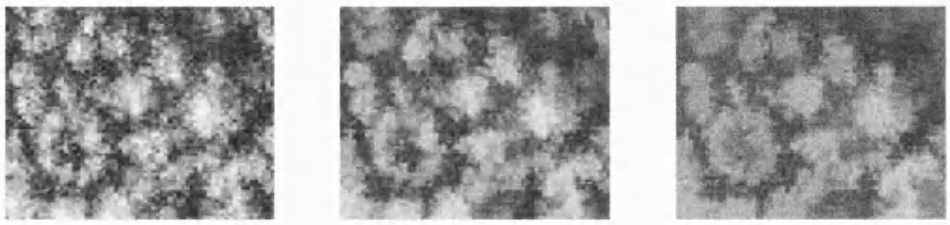
(a) Original signal.

(b) Area open-close.

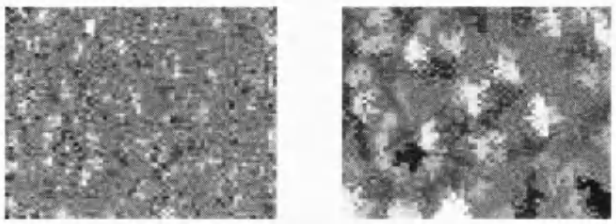
Figure 2.16: An example of area morphology.

Area morphology can be used to describe a set of SE's to be used in granulometric analysis of images, see Section 2.5.1. Based on the HVS, large structures contain more information than smaller structures. Therefore, area morphology can be used for the multi-scale decomposition of images [26]. The granulometric datasieve in Figure 2.14, can be constructed using area morphology to describe the set of SE's.

An example of a datasieve based on area morphology is shown in Figure 2.17. The original image, Figure 2.17(a), is a 64×64 pixel region of trees taken from a SAR image, and has been processed by area open-close operations with area sizes $\alpha = 16$ and $\alpha = 64$. The grain images in Figures 2.17(d)-(e) contain both positive and negative values, therefore the mid grey-level of the images should be considered to be zero. At an area size $\alpha = 16$ only small scale features are present in the grain image. However the grains at area size $\alpha = 64$ show the peaks and troughs between individual trees, showing how the granulometric decomposition of an image can be used to extract textural features.



(a) Original image. (b) Sieve $\alpha = 16$. (c) Sieve $\alpha = 64$.



(d) Grains $\alpha = 16$. (e) Grains $\alpha = 64$.

Figure 2.17: An example of granulometric decomposition using area morphology.

2.5.3 Morphological Segmentation

The previous sections have described how mathematical morphology can be employed to analyse features within an image dependent on their size or *scale*. However, mathematical morphology can be utilised to identify varying scale regions within an image [16], providing a multi-scale image analysis tool.

When an image is interpreted as a topographic map, mathematical morphology provides a method for segmentation [48]. In the topographic representation of an image, the numerical value of each pixel represents the elevation of the surface at the given point. Using the topographic surface, if a theoretical drop of water was to fall on the surface, it would flow down the steepest slope until a local minimum was reached [47]. The set of points on the topographic surface whose steepest descent reaches a given minimum establish the *catchment basin* associated with the corresponding minimum, Figure 2.18. The *watersheds* are the regions which separate two adjacent catchment basins.

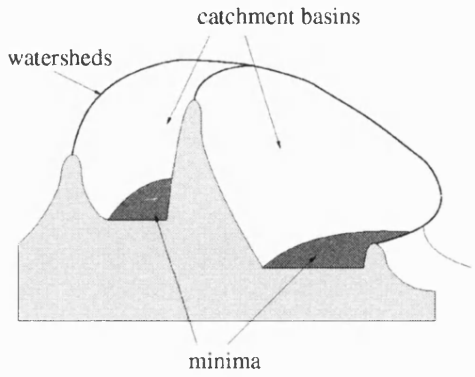


Figure 2.18: Watersheds of a topographic surface.

The watershed transform [16] is a mathematical morphology tool used for the purpose of image segmentation. A suitable algorithm, based on immersion simulations [139], can be used for the calculation of watersheds in grey scale images. The algorithm is based on an immersion process analogy, where the image is flooded with water, simulated using a queue of pixels. The algorithm is split into two stages. Firstly the pixels are sorted in the increasing order of grey values, providing a method for direct access to pixels at a given water level. Once the pixels have been sorted, the catchment basins of the image are immersed, simulating the process of flooding. If the flooding has been done to a given level h, every catchment basin whose corresponding minimum has an altitude lower than or equal to h will have a unique label. Due to the initial sorting stage all pixels of altitude $h+1$ and higher can be directly accessed. The pixels higher than $h+1$ which have a labelled pixel in the local neighbourhood are placed in a queue. This queue structure enables the labelled catchment basins to extend, leaving only minima at level $h+1$.

If the image objects are considered as regions of homogeneous grey values, the detection of high grey level variations can be used to locate the object boundaries or edges. Gradient operators are used to enhance the edges of an image. The basic morphological gradient, also called *Beucher gradient* [16], is defined as the arithmetic difference between the dilation and erosion

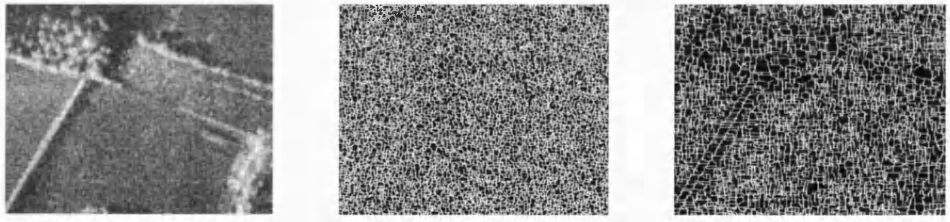
$$\rho_B(X) = \delta_B(X) - \epsilon_B(X). \tag{2.24}$$

Therefore, when the main criterion for segmentation is the grey value homogeneity of the objects in the image, the watershed transform is applied to the gradient image. However, when an image is corrupted by noise, applying a gradient operator can enhance the noise component [124]. The watershed transform of the gradient image results in apparently homogeneous regions fragmented into small segments [16].

For texture segmentation the regions of interest do not consists of homogeneous grey values. Also in the presence of multiplicative speckle noise, Section 1.2.2, gradient operators detect more false edges [131]. The coefficient of variation (CV) is computed as a texture measure

$$CV = \frac{1}{SNR} = \frac{\sigma}{\mu}, \tag{2.25}$$

where μ and σ are the mean and standard deviation respectively of a local neighbourhood. A large CV value indicates the presence of an edge, or texture boundary. Hence for texture segmentation the watershed algorithm can be applied to the CV image [146]. Results of the watershed transform when applied to the morphological gradient image and CV image are presented in Figure 2.19.

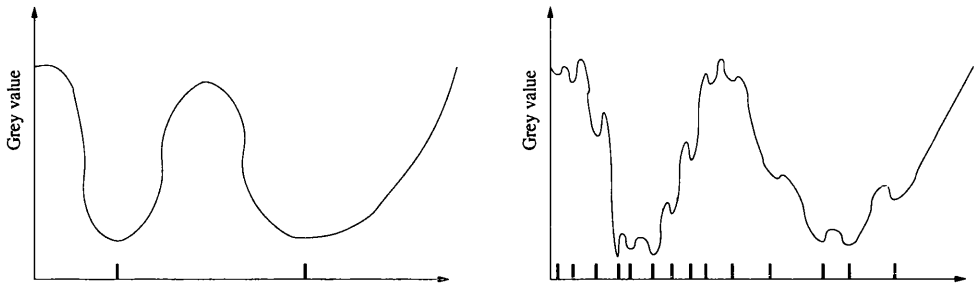


(a) Original image. (b) Grad segmentation. (c) CV segmentation.

Figure 2.19: An example of watershed segmentation.

Analysis of Figure 2.19 shows one problem of the watershed transform is over-segmentation, which is exaggerated in the presence of noise. This is because the presence of noise can increase the number of local minima of a signal, see Figure 2.20. An increase in the number of local minima in an image increases the number of catchment basins in the immersion process and leads to an over segmentation of the image.

The watershed transform can be applied at multi-levels and will merge fragmented



(a) Clean signal. (b) Noisy signal.

Figure 2.20: Local minima on two similar signals.

regions. This is because the boundaries identified from the segmentation have different weights. Boundaries which are within a homogeneous region are weaker [16]. In order to compare the boundaries a relationship between the neighbourhoods needs to be introduced. This can be achieved by the *mosaic image* [16]. After the segmentation of an image I, every catchment basin is labelled with the grey value corresponding to the minima of $\rho_B(I)$. The result is a simplified image, made up of a mosaic of pieces (catchment basins), of constant grey values. In a mosaic image no information regarding the contours has been lost. The calculation of the mosaic image has been used to process X-ray images prior to the watershed segmentation [16].

Another method to reduce over fragmentation in the segmented image is to evaluate the minima on a criterion based on contrast: the dynamics [53]. To define how dynamics can be used to reduce the number of catchment basis in an image, the following definitions need to be introduced.

Path on an image: A path $P(x, y)$ on an image is defined as a set of adjacent pixels $p_1, p_2, p_3, \ldots p_n$ that links pixel x to pixel y. $P(x, y) = \{p_1, p_2, p_3, \ldots p_n\}$ such that:

- p_1 = pixel x and p_n = pixel y.
- For any $i \in [1, n]$, p_i and p_{i+1} are neighbouring image pixels.

A path is defined as direct if it does not pass through through any pixel twice.

Dynamics of a path: The dynamics of a path, $P(x, y)$ that links pixel x and pixel y on an image I, is the difference between the points of highest and lowest of altitude on the path.

$$\text{Dyn}(P(x,y)) = \{\sup(|I(x_i) - I(x_j)|), x_i \in P(x,y), x_j \in P(x,y)\} \quad (2.26)$$

The dynamics of a path $P(x, y)$ on a 1-D signal is illustrated in Figure 2.21.

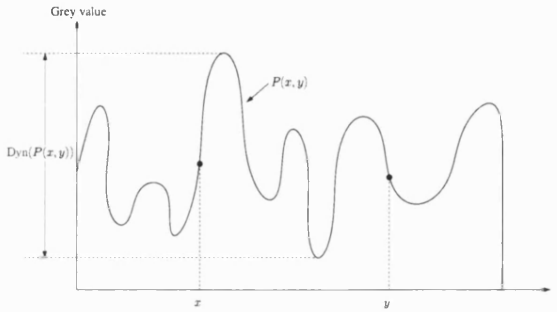


Figure 2.21: Dynamics of a path.

Dynamics of a minimum: The dynamics of a minimum M is equal to the dynamics of the path with the lowest dynamics that links the minimum M to a pixel y that belongs to the catchment basin, C, whose minimum is lower in altitude than M.

$$\text{Dyn}(M) = \{\inf(\text{Dyn}(x,y)), x \in M, y \in C : \text{alt}(M) > \text{alt}(\min(C))\} \quad (2.27)$$

This definition is illustrated in Figure 2.22.

Once the dynamics for all the minima have been calculated for an image, they can be used to assign a weighting to each minima. For a noisy signal the dynamics of most of the minima is low, indicating these minima correspond to non-significant structures [53]. The minima of an image can be identified by a marker which is the dynamic value associated with each minima. By setting a threshold to the markers, this will merge fragmented regions due to noise. An example of the *hierarchical* segmentation [16] is shown in Figure 2.23. By setting a threshold to the markers minima due to noise components are removed, providing a multi-scale segmentation.

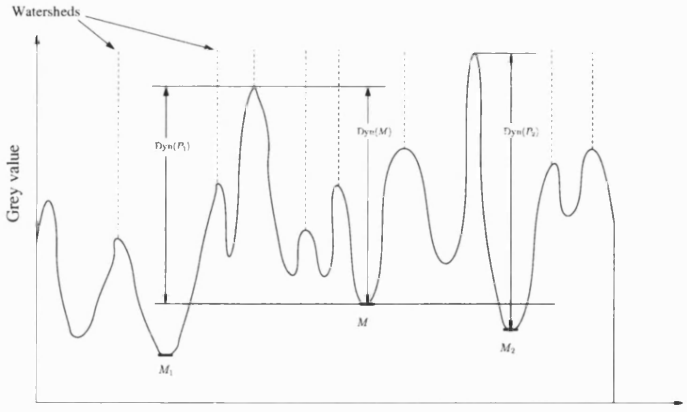


Figure 2.22: Dynamics of a minimum.

Figure 2.23: An example of multi-scale watershed segmentation using markers.

2.6 Discussion

Investigations into the function of cortical cells imply that image representation in both the spatial and spatial frequency domains can model biological vision systems [114]. Analysis of the cortical cells tuning characteristics indicate that the receptive field of cells are confined to both spatial location and spatial frequency bandwidth.

In this chapter three multi-scale image analysis techniques have been introduced. Firstly Gabor functions have been studied, which are both bandlimited and localised in space, and provide a suitable representation for the human vision system [98]. A bank of Gabor filters can be utilised to extract image features at different scales and orientations, which are suitable for texture analysis.

The second multi-scale image analysis technique studied is the wavelet transform. Wavelet analysis is applied to digital images with the discrete wavelt transform, DWT, implemented with a two-channel filterbank. Traditionally the DWT is applied recursively to the low frequency approximation of the input image. However, it has been proven that textural information resides in mid-frequency regions of an image, giving rise the the wavelet packet transform. By implementing a full wavelet packet decomposition suitable texture features can be extracted from an image and used for texture classification.

Finally a non-linear set of operators, namely mathematical morphology, are introduced. Morphological operators extract image information based on shape, thus yielding features that are directly related to objects within an image. By applying increasing scale morphological operators iteratively to an image a data sieve is constructed. The outputs of the data sieve are analogous with linear band-pass and low-pass filters, providing an alternative to linear filterbanks for texture feature extraction. The final sections of this chapter study multi-scale segmentation via the morphological watershed transform. The watershed transform can be utilised to identify local regions of an image that are to be classified based on their textural features.

Chapter 3

Speckle Reduction

THE product model for SAR clutter introduced in Section 1.2.1 introduced the principal that the observed intensity in a radar image comprises of an underlying RCS on which speckle is imposed. Speckle has the effect of reducing the image contrast and making fine detail indistinct, making image interpretation a more difficult task. Texture in SAR images can be separated into two components [133]:

1. Texture due to scene variations.
2. Texture due to speckle (multiplicative correlated noise).

Therefore, when interpreting a SAR image on the basis of texture, it is important that the presence of speckle does not effect the texture measure. This leads us to the problem of deriving the RCS, given the observed intensity of the image. Speckle suppression techniques fall two into categories. In the first category, speckle is smoothed by averaging a number of images of the same scene, with independent speckle patterns. Speckle removal by these methods are accomplished in the pre-image formation stage [73]. Techniques in the second category suppress speckle by applying filtering techniques in the post-image formation stage. The aim of filtering techniques is to smooth homogeneous regions while retaining edge information [89].

This chapter investigates existing techniques for the removal of speckle from SAR images. Section 3.1 describes how speckle can be reduced by using data from the same scene recorded at different times. However, this technique has the disadvantage of reducing the spatial resolution of the image. Section 3.2 uses the theoretical models of SAR data to recover the underlying RCS from the speckle noise. As the emphasis of this research is on a multi-resolutional approach to textural feature extraction, it is multi-resolutional speckle reduction techniques that will be investigated in most depth. The properties of the discrete wavelet transform that make it suitable for noise reduction is described in Section 3.3. Existing algorithms that utilise the wavelet transform for speckle noise reduction in SAR images are studied in Section 3.3.1 and Section 3.3.2. Non-linear mathematical morphology operations for the use of noise reduction are studied in Section 3.4. Operations based on the use of a single structuring element are described in Section 3.4.1. These techniques have the effect of adding artefacts to the processed image, which can be reduced by the use of soft morphology, Section 3.4.2, and area morphology, Section 3.4.3. Multi-scale morphological speckle reduction techniques are studied in Section 3.4.4, which leads to the novel texture preserving granulometric shrinkage algorithm described in Section 3.4.5. This chapter is concluded by a comprehensive evaluation of speckle reduction techniques in Section 3.5. The evaluation techniques include quantitative measures including the reduction in standard deviation in Section 3.5.1, and preservation of textural information, Section 3.5.3.

3.1 Multi Looking

Independent images of the same scene can be acquired by splitting the original system bandwidth into several bands in the pre-image formation stage. When the subimages (looks) are given the same weight and are uncorrelated they exhibit the properties of non-overlapping spectra [4] [99].

The signal-to-noise ratio SNR is defined as the mean intensity over the standard deviation. From (1.4) for single look images this value is unity. However, the averaging of L independent looks will preserve the mean value r while reducing the the variance

by a factor L to become $\frac{r^2}{L}$. Therefore the SNR will be improved by a factor of $\sqrt{L}$

$$\frac{SNR_o}{SNR_i} = \frac{r}{\sqrt{\frac{r^2}{L}}} = \sqrt{L}. \tag{3.1}$$

If there is overlap between the subbands, correlation exists and the number of looks used to generate the image is reduced. The effective number of looks, L_{eff}, is the number of independent intensity values averaged per pixel [142], and is defined by

$$L_{eff} = \frac{\text{mean}^2}{\text{variance}}. \tag{3.2}$$

Another approach to improving estimates of the RCS is to assume r is constant over a neighbourhood surrounding a given pixel. Given a window of L independent pixel values, this is the same situation as described above, where independent looks of the same point can be averaged. This method of multi-look despeckling fails where the assumption of constant RCS within the window breaks down [96]. Only if features of interest within the scene occur on a large enough scale can multi look despeckling be effective. The speckle suppression achieved by multi-look techniques is at the expense of spatial resolution and limits the number of looks used to form the image.

3.2 Filtering Techniques

The aim of filters for speckle reduction is to derive the underlying RCS r given an image of intensity values. Filters seek to replace the central pixel of a local region with an estimate of r, obtained from those pixels within the neighbourhood. Image filtering techniques are applied to pre-formed images with the aim of suppressing speckle whilst retaining image detail.

3.2.1 Adaptive Speckle Filters

Speckle reduction techniques involving averaging pixels within a window centred on the pixel being processed reduce spatial resolution, having the effect of blurring edges and smearing fine detail. Lee proposed the sigma filter [73, 72] which is implemented to reduce local noise. This is achieved by selecting a sub-region within the window with the greatest uniformity. The sigma filter is a local statistics method and excludes pixels in the neighbourhood where the grey level is significantly different from the pixel being processed. In image smoothing any pixel outside the range of two standard deviations from the pixel being processed most likely comes from a different population and therefore should be excluded from the average process. The probability of a sample falling within two standard deviations of a Gaussian distribution is 0.955, meaning 95.5% of random samples are within the specified range. In a $(2n+1)(2m+1)$ window centred on pixel (i, j) let

$$\delta_{k,l} = \begin{cases} 1 & \text{if } I_{i,j} - 2\sigma_n \leq I_{k,l} \leq I_{i,j} + 2\sigma_n \\ 0 & \text{otherwise,} \end{cases} \tag{3.3}$$

where σ_n is the standard deviation of the neighbourhood. An estimate of the RCS at pixel (i, j) can be expressed as

$$\hat{r}_{i,j} = \left(\sum_{k=i-n}^{i+n} \sum_{l=j-m}^{j+m} \delta_{k,l} I_{k,l} \right) \left(\sum_{k=i-n}^{i+n} \sum_{l=j-m}^{j+m} \delta_{k,l} \right)^{-1}. \tag{3.4}$$

An advantage of the sigma filter is that the range of intensity values included in the average are not fixed thresholds and vary according to the intensity of the centred pixel, having the advantage of edge preservation.

A filter based upon reconstructing a SAR image from the underlying Gamma distribution is presented in [38]. The filter operates by estimating the best fit Gamma distribution to each image pixel. The distribution is determined by calculating estimate scale and shape parameters, $\hat{\alpha}$ and $\hat{\beta}$, from intensity values within a local neighbourhood using a maximum likelihood method. The filter's output is then the value with the

highest probability of the found Gamma distribution

$$\Gamma_{ml} = \hat{\beta}(\hat{\alpha} - 1). \tag{3.5}$$

A better representation of the underlying pdf can be achieved if the effective number of looks is estimated. A maximum likelihood estimate for $\hat{\alpha}$ is given by

$$\log \hat{\alpha} - \Psi(\hat{\alpha}) = \log\left(\frac{1}{N}\sum_{n=1}^{N} I_n\right) - \frac{1}{N}\left(\sum_{n=1}^{N} \log I_n\right). \tag{3.6}$$

where $\Psi(\hat{\alpha})$ is the digamma function. A solution for $\hat{\alpha}$ can be found using numerical techniques such as a Newton-Raphson iteration method. As $\overline{I} = \hat{\alpha}\hat{\beta}$, once $\hat{\alpha}$ has been determined, $\hat{\beta}$ can be calculated using

$$\hat{\beta} = \frac{1}{\hat{\alpha}N}\sum_{n=1}^{N} I_n. \tag{3.7}$$

3.2.2 Bayesian Reconstruction

Before the discussion of probability theory it is important to clarify the notation that will be used. Table 3.1 shows basic probability definitions.

$P(A)$	Probability of event A.
$P(AB)$	Probability of events A and B.
$P(A\|B)$	Probability of event A given event B.

Table 3.1: Basic probability definitions.

Speckle reduction of a SAR image can be achieved in terms of Bayes theory relating the observed intensity I, to the RCS r. The basic foundation of probability theory follows the definition of conditional probability.

$$P(Ir) = P(I|r)P(r) \tag{3.8}$$

In (3.8) the events I and r are simultaneous and have no temporal order, therefore

$$P(Ir) = P(rI) = P(r|I)P(I). \tag{3.9}$$

By equating (3.8) and (3.9) this leads to the common form of Bayes theory

$$P(r|I) = P(I|r)\left(\frac{P(r)}{P(I)}\right), \tag{3.10}$$

where $P(I|r)$ is the likelihood function, which describes the effect of speckle on the image, for multi-look images this is given by (1.7). $P(r)$ and $P(I)$ are known as *priors* that encapsulate prior knowledge about the RCS and intensity values. Estimating the RCS value r that is derived from the observed value I, is equivalent to maximising the likelihood function.

An estimate the RCS value r can be derived from the observed value I if the form of the PDF $P(r)$ is known, this is referred to as the *maximum a posteriori* MAP estimate [68]. It has been shown in Section 1.2.3 that the RCS of natural clutter can be represented by the Gamma distribution (1.8).

When both likelihood (1.7) and priori (1.8) PDFs are known, the PDF of r given an intensity I is available [93],

$$P_{AP}(r|I) \propto P(I|r)P(r) = \frac{L}{r\Gamma(L)}\left(\frac{LI}{r}\right)^{L-1}\exp\left(-\frac{LI}{r}\right)\frac{r^{\alpha-1}}{\beta^{\alpha}\Gamma(\alpha)}\exp\left(-\frac{r}{\beta}\right), \tag{3.11}$$

where L is the number of looks and Γ is the gamma function. Hence, the log likelihood is given by

$$\begin{aligned}\lambda &\equiv \ln P(I|r) + \ln P(r) \\ &= L\ln(L) - \ln(r\Gamma(L)) + (L-1)\ln(LI) - (L-1)\ln(r) - \tfrac{LI}{r} \\ &+ (\alpha-1)\ln(r) - \ln(\beta^{\alpha}\Gamma(\alpha)) - \tfrac{r}{\beta}.\end{aligned} \tag{3.12}$$

The corresponding *MAP* solution for r can be found by finding the maximum of the log

likelihood function. Differentiating (3.12) with respect to r [96] gives the quadratic,

$$\frac{1}{\beta}r_{MAP}^2 + (L + 1 - \alpha)r_{MAP} - LI = 0, \tag{3.13}$$

where α and β are estimated from the local neighbourhood.

3.3 Discrete Wavelet Transform Speckle Reduction

The principal of removing additive Gaussian noise from signals, via the discrete wavelet transform, was introduced by Donoho [31]. It was stated that a clean signal is represented by only a few large wavelet coefficients. Gaussian noise is evenly distributed throughout the wavelet coefficients with a relatively small magnitude. This leads to the theory of applying a threshold to the wavelet coefficients, thus removing the contribution from the Gaussian noise while preserving image detail [71].

Although the speckle noise for SAR images is a multiplicative negative exponential distribution, Arsenault [8] shows that for logarithmically transformed intensity image the speckle noise is approximately Gaussian Additive noise. Figure 3.1 shows the histogram of a logarithmically transformed simulated SAR image and the best-fit Gaussian distribution. A measure of the similarity between two distributions can be obtained via the *chi-square test*, see Section E. As the comparison is between two binned data sets the chi-squared value was calculated with (E.2) resulting in $\chi^2 = 0.0683$. The value of the *chi-square probability function* (E.3) is close to zero, indicating the probability that the distributions are not from the same population is very small.

3.3.1 Wavelet Shrinking

Guo *et al.* [54] proposed a speckle reduction method based on thresholding the wavelet coefficients of a logarithmically transformed image. The method provides significant speckle reduction while not altering the resolution of the original SAR imagery. The preservation of resolution has the effect of protecting edge information and not

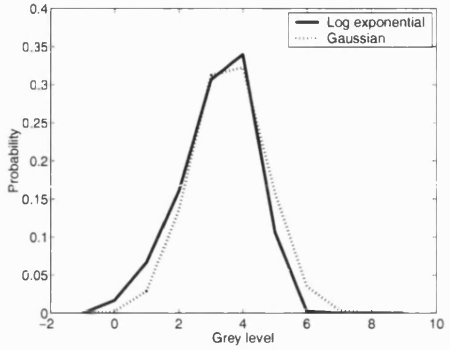


Figure 3.1: Approximate Gaussian of log intensity values.

smoothing fine image detail. Therefore it can be used as a pre-processing stage to improve image interpretation based on texture.

For an image of the form given by (1.3), Figure 3.1 shows that for a logarithmically transformed SAR image, the speckle is approximately Gaussian additive noise, *i.e.*

$$\tilde{I} = \tilde{r} + \tilde{n} \tag{3.14}$$

where $\tilde{I} = \ln |I|$. If I represents the original input image and W is the multi-level DWT, then a multi-level representation is given by $y = W(I)$ (or $\tilde{y} = W(\tilde{I})$). The noise level in the multi-resolution representation is not known a priori and therefore has to be estimated. A suitable estimate of σ is taken to be the standard deviation of the HH subband of the first level of the DWT [130]. Once the original input image has been transformed into its multi-resolution representation, a formula can be applied to the subband images to accomplish speckle reduction and enhancement of SAR images. Let f be a non-linear function designed to reduce speckle in the wavelet subbands. Then an enhanced subband image $\hat{y}$ is given by

$$\hat{y} = f(y) \tag{3.15}$$

The speckle reduced image is then obtained from the inverse DWT of the enhanced subband image $\hat{y}$. The complete wavelet shrinkage process is given in Figure 3.2.

There is a choice of two thresholding functions that can be used, namely hard and

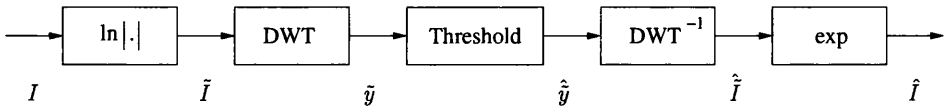
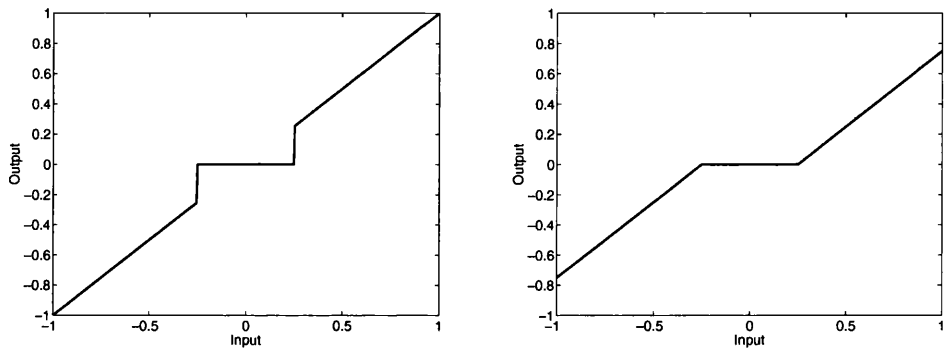


Figure 3.2: Wavelet shrinkage process.

soft thresholding as defined in Table 3.2. The visualisation of the thresholding functions, Figure 3.3, shows that the soft thresholding function is more appropriate at suppressing additive Gaussian noise which is evenly distributed throughout the wavelet coefficients. An appropriate value for the threshold t is $\gamma\sigma$ where γ is a constant.



(a) Hard thresholding.

(b) Soft thresholding.

Figure 3.3: Thresholding functions.

Hard Thresholding	Soft Thresholding.
$f(\tilde{y}) = \begin{cases} \tilde{y} & \text{if } \lvert\tilde{y}\rvert > t \\ 0 & \text{otherwise} \end{cases}$	$f(\tilde{y}) = \begin{cases} \tilde{y} - t & \text{if } \tilde{y} > t \\ 0 & \text{if } \lvert\tilde{y}\rvert \leq t \\ \tilde{y} + t & \text{if } \tilde{y} < t \end{cases}$

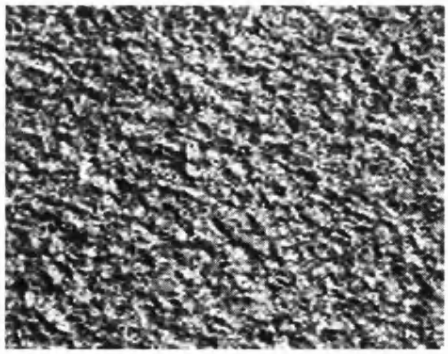
Table 3.2: Thresholding functions.

The thresholding scheme for SAR images proposed in [54] differs from that proposed by Donoho [31] by not thresholding the LL subband of the final level of decomposition. This guarantees the mean intensity of the processed image is the same as the original input image.

Results from a wavelet shrinkage technique are presented in Figure 3.4. An original image taken from Brodatz texture album [22], is corrupted with 3-look Gamma distributed single point speckle (1.7). Although the speckle reduced image, Figure 3.4(c), looks similar to the original image, a full evaluation of the effectiveness at removing the speckle noise can only be obtained by quantitative evaluation, which is considered in Section 3.5.

(a) Original textured image.

(b) Image corrupted with 3 look SAR speckle.

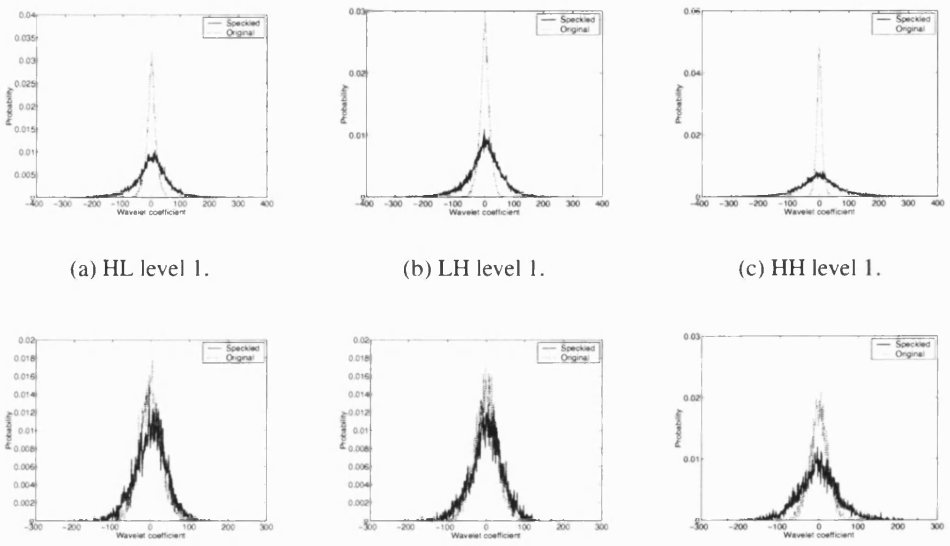


(c) Level 1 speckle reduction.

Figure 3.4: Soft threshold wavelet shrinkage speckle reduction results.

The original image is restored by applying a soft thresholding technique to the wavelet coefficients at the first level of the wavelet pyramid decomposition. It is only necessary to apply the wavelet shrinkage at the first decomposition level, as the speckle was

introduced on a single pixel level. The correlation properties of speckle have an effect up to a scale that corresponds to the granular size [115], which for single point speckle corresponds to the first decomposition level. Histograms of the wavelet coefficients are shown in Figure 3.5. This reinforces the fact that single point speckle is only present in the first level wavelet coefficients. A value of $\gamma = 2$ was chosen for the threshold. The method of wavelet shrinking to remove the speckle noise component from SAR



(a) HL level 1. (b) LH level 1. (c) HH level 1.

(d) HL level 2. (e) LH level 2. (f) HH level 2.

Figure 3.5: Wavelet coefficient histograms.

imagery has been widely applied [90] [130] [54].

3.3.2 Wavelet Coefficient Range Reduction

Fukuda and Hirosawa [45] proposed a method of speckle reduction in the wavelet domain by reducing the range of the pixel power in each wavelet subband. As shown in Table 2.1, there are three detail images produced by the wavelet transform each containing different information about the input image. If the power range of the entire detail image is reduced this will not only smooth speckle but also edges, introducing

aliasing. It was proposed that for a high value in each of the detail images the local neighbourhood should be examined to determine whether the pixel is an edge or corrupted by speckle. The local neighbourhood in the different subbands are shown in Figure 3.6.

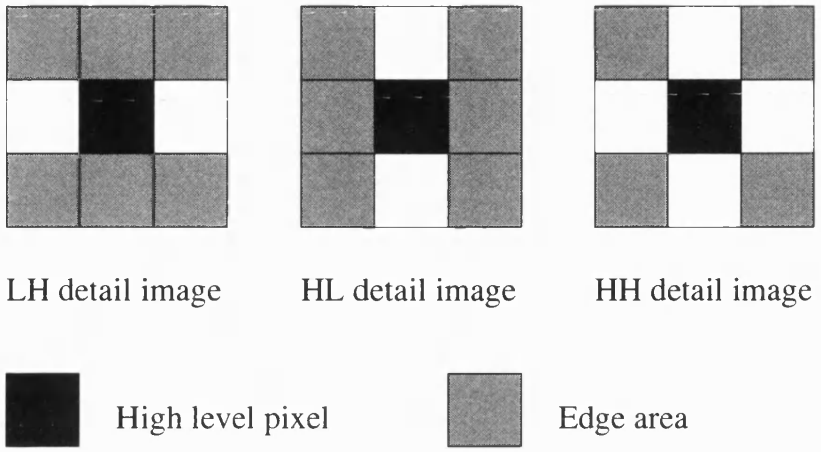


Figure 3.6: Neighbourhoods for judgement of edge information.

The proposed algorithm to suppress speckle while preserving edge information is as follows.

1. Classify each pixel of the detail images as either high or low level using the threshold T.
2. Reduce the power of low level pixels to $\alpha\%$.
3. Consider a 3×3 window around any high level pixels and define the edge area for the given high level pixel, as shown in Figure 3.6. If the edge area in the window contains at least one high level pixel, the centre pixel is assumed to contain edge information and keep its original value.
4. If no pixels in the edge area are high level pixels, the centre pixel is regarded as speckle component, and its power is reduced by $\beta\%$.

3.4 Morphological Speckle Reduction

Linear techniques for noise reduction replace each pixel with a function of the pixel's neighbourhood. As the 1-D features of an image have common frequency components to noise, they are not separable in the frequency domain [97]. Therefore, when linear filters are applied to an image for the removal of noise, they have the effect of blurring edges and image features. Median filters [5] are rank order filters that behave in a similar manner to low-pass filters. Median filters can be employed to remove impulse noise whilst preserving edges. However, such filters perform poorly in regions of high density spatial noise [125].

Mathematical morphology provides operators suitable for the task of noise removal in images and has been used in the field of image processing for the enhancement of images [42], the removal of noise from binary images [108]. The removal of dirt and scratches from archived film has been achieved by utilising mathematical morphology operators [58]. As the corruption is only present in the luminance fields of a video sequence, only the luminance frame is filtered. This demonstrates that morphology based noise removal filters are suitable for the processing of grey-scale images. These techniques have been extended to images from coherent systems with the removal of speckle noise in ultrasound images [125] [9] and SAR images [109] [65] [87] [106].

3.4.1 Structuring Element-Based Morphology

During a study of textural analysis of sea-ice in SAR imagery [60], a filtering algorithm was developed that reduced the effect of speckle whilst preserving edge definition. Homogeneous regions of multi-year ice fields appear to have uniform intensity on optical images. However, the same regions monitored by SAR appear speckled. The presence of speckle presents major difficulties to pixel-by-pixel classification algorithms. An algorithm was developed to reduce the effects of speckle without degrading edges between the different types of ice.

The algorithm processes the image in the terms of the underlying image algebra. This provides a means of probing an image in terms of other images, or *structuring el-*

ements. The technique developed is a nonlinear isotropic[1] filtering algorithm based on a pixel's neighbourhood. Isotropic filtering consists of alternately calculating the maximum value over a disk of radius r, followed by calculating the minimum value over a disk of radius $r + 1$. This is done for every point in the image and terminates when the radius of the disk produces *unbiased* results. The choice of structuring elements were disk-shaped because they treat all directions equally. It was discovered that disks of small radii were effective at removing noise, whilst disks of larger radii remove deviations from the local average, giving an approximation to the mean value of individual ice types. One of the problems with morphological filters based on SE's is the introduction of geometrical and topological distortion introduced by the choice of SE [108]. The open-close operator (D.12) has been used to remove speckle noise in SAR images as a pre-processing stage to watershed segmentation [146].

3.4.2 Soft Morphology

Soft morphological operations are non-linear processes related to the class of basic morphological operations, see Appendix D. The aim of soft morphology is to relax the definitions of standard morphological operations to improve *robustness*, whilst maintaining the desirable properties of basic morphological operations. The concept of soft morphology is the SE is split into 2 parts:

1. A hard centre which behaves as a standard SE.
2. A soft boundary where the minimum and maximum operations of standard morphology are replaced by alternative order statistics.

The result of soft morphology is the operations behave less *harshly* in noisy image regions, and are more tolerant to finer details of image features. Soft morphology has been used for the removal of dirt and scratches from archived film [58] and the suppression of speckle noise in SAR imagery [87].

[1] having physical properties that do not vary with direction.

Morphological operators based on a structuring element and boundary can be used for the removal of speckle from SAR images [65]. The new operators are boundary erosion region dilation ($BERD$) (3.16) and boundary dilation region erosion ($BDRE$) (3.17).

$$BERD = I\boxminus(D, B) = (I \ominus B) \oplus D \tag{3.16}$$

$$BDRE = I\boxplus(D, B) = (I \oplus B) \ominus D \tag{3.17}$$

An example of (D, B) used in [65] is shown in Figure 3.7. The examples in Figure 3.7 are arbitrary *e.g.* B could be wider than one pixel.

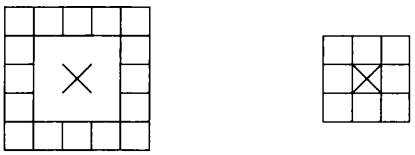


Figure 3.7: An example of (D,B) structuring elements.

The morphological operators used to construct the filter in [65] are based on the BERD and BDRE operators. The two new operators are the bounded closing (3.18) and bounded opening(3.19) of I with (D, B).

$$I\boxdot(D, B) = \max\{I, I\boxminus(D, B)\} \tag{3.18}$$

$$I\boxdot(D, B) = \min\{I, I\boxplus(D, B)\} \tag{3.19}$$

The bounded opening and closing operations act as filters to remove small noisy regions from an image. To remove speckle a sequence of several operations, called *alternating sequential filters* or ASF [106], are applied to the image. The k^{th} order

ASF for an image I with SE B is given by:

$$\text{ASF}\{I \bullet B \circ B\}^k = I \bullet B \circ B \bullet B^2 \circ B^2 \ldots \bullet B^k \circ B^k. \tag{3.20}$$

The SE B^k is obtained by:

$$B^k = B \oplus B \oplus B \oplus B \oplus B \ldots k \text{ times}. \tag{3.21}$$

The ASF used for the removal of speckle in [65] is:

$$\text{ASF}\{I \boxdot (D,B) \boxdot\!\!\boxdot (D,B)\}^k = I \boxdot (D,B) \boxdot\!\!\boxdot (D,B) \boxdot (D^2,B^2) \boxdot\!\!\boxdot (D^2,B^2) \ldots k \text{ times}, \tag{3.22}$$

where B^m is the boundary of SE D^m. The basic element D used in (3.22) is a square of 3 unit lengths [65].

The concept of image analysis with two structuring elements is also utilised in the morphological *hit-or-miss* transform [51] [126]. By defining a pair of disjoint sets, $B = \{B_1, B_2\}$, or composite SE, the hit-or-miss transform for an image I is given by,

$$I \circledast B = (I \ominus B_1) \cap (I^c \ominus B_2), \tag{3.23}$$

where I^c is the complement of I. As I^c is an infinite set then the local background of I must be bounded by a window W, defined as the set difference $(W - I^c)$, resulting in the transform,

$$I \circledast B = (I \ominus B_1) \cap ((W - I^c) \ominus B_2). \tag{3.24}$$

3.4.3 Area Morphology

The morphological operators NOP and NCP [26] based on the area of connected components can be used for noise reduction. Area morphology has the advantage over standard morphological operators as it will preserve structures of any given shape, and will not create any artificial artefacts due to the choice of a single SE [26].

Area morphology has been used in the study of a microscopy images of a metal al-

loys [137]. The microscopy images contained intra-grain texture that had been corrupted by noise. When the image was processed using standard morphological operators and a square SE of 9 pixels, most of the inter-grain lines were disrupted whilst still preserving some of the noise components. Area morphological operations, Section 2.5.2, were used to process the images and resulted in noise removal whilst preserving inter-grain separations.

3.4.4 Multi-Scale Morphological Speckle Reduction

The noise removal properties of morphological openings and closings are limited by their tendency to remove thin features along with image noise [97]. The morphological image cleaning algorithm (MIC) [97] is a noise removal algorithm that preserves thin features. The MIC algorithm operates by processing residual images – the difference between the original image and morphologically smoothed images. Residuals at a number of different scales are calculated via a morphological size distribution. Regions in the residual images which are judged to be noise are removed. A de-noised image is generated by reconstructing the residual images with the morphologically smoothed version. The MIC algorithm morphologically smooths an image with a pixel-wise average of the open-close and close-open of I,

$$OCCO(I;B) = \frac{1}{2}(I \circ B) \bullet B + \frac{1}{2}(I \bullet B) \circ B, \tag{3.25}$$

where I is the image to be processed and B is an SE.

The choice of SE B in [97] are disk-shaped with increasing radii. The image is filtered with a range of SE sizes, so the various residuals can be processed separately. The image is filtered with successively $OCCO$ filters until one of the smoothed images, at diameter d, appears to contain no noise component. The previous images (with smaller SE diameters) therefore contain noise components and should be processed. Smoothing the image with several $OCCO$ filters at different diameters results in a morphological pattern spectrum [80]. If S_α is the result of smoothing an image I with an $OCCO$ filter of diameter d_α. Let $S_0 = I$ since $B_{d_0} = 1$, *i.e.*

$$OCCO(I;B_{d_0}) = I, \tag{3.26}$$

then each D_α is a residual image that constitutes a *size band*,

$$D_\alpha = S_{\alpha-1} - S_\alpha. \tag{3.27}$$

The residual images contain both noise and image features. In cases where the standard deviation of the noise is smaller than the amplitudes of the image features, at a given size α, the noise in D_α can be removed by thresholding.

A method of removing speckle noise from ultrasound images whilst preserving image features is presented in [125]. The input image is decomposed by generating residual images, using a series of morphological closing operations employing different SE sizes. Each residual image contains both speckle noise and feature components. In the residual images the amplitudes associated to the speckle noise are lower than the image features [125]. This property is analogous with the wavelet transform decomposition of images [31] and means that it is possible to separate the noise components from image features using a histogram of each residual image. Once a suitable threshold value has been selected, the noise can be removed whilst preserving the image features.

3.4.5 Granulometric Shrinkage

The granulometric shrinkage speckle reduction technique [40], is developed as part of this thesis. If an image is decomposed using a datasieve, Section 2.5.1, the grain (or residual) images will contain both image features and noise components. An area morphological scale-space described by (2.21) can be thought of as being analogous with a multiscale wavelet transforms. Here, the sieved image S_λ is the lowpass image and the granule images G_i at scales $i = 1, 2, \ldots, \lambda - 1$ are the bandpass detail images. The grain images are signed images as each G_i is the difference between two area morphology filters. The zero regions of G_i identify regions where S_{i-1} and S_i coincide. Positive regions mark where S_{i-1} is brighter than S_i and in negative regions the greyscales of S_{i-1} are lower than those in S_i.

It is known that the logarithmic transform of SAR data results in the speckle noise component approximating additive Gaussian noise, see Section 3.3. Therefore it can be as-

sumed that the positive and negative values in the grain images can be processed in the same manner as the wavelet shrinkage algorithm, see Section 3.3.1, to achieve noise removal. Noise reduction via granulometric shrinkage can be achieved by replacing the forward and inverse wavelet transform in Figure 3.2 by an area morphology granulometric sieve, and employing an appropriate thresholding function. The underlying assumption behind this approach is that the positive and negative values in the grain images can be processed in the same manner as wavelet coefficients to achieve noise removal. However, as the datasieve has edge preserving properties, this approach has the potential of reducing speckle noise with less texture modification than the wavelet approach.

To remove noise, the soft thresholding function shown in Table 3.2 can be used, providing that a suitable estimate of the noise level can be found. The grain image G_1 contains single pixel features which, being too small to be image structures, can be assumed to be noise components. Therefore, the standard deviation of G_1, when scaled by a constant, provides a suitable threshold value.

3.5 Evaluation of Speckle Reduction

A description of a good speckle noise filter is one which filters homogeneous regions, whilst respecting edges and textural information [36]. A visual assessment of the speckle reduced images can help in the evaluation of the filtering algorithm. However, visual quality assessment is a psycho-physical process and therefore a subjective measure. A quantitative measure can be obtained by measuring the mean value, $\overline{x}$, the standard deviation σ_x, and the signal-to-noise ratio (SNR) $\overline{x}/\sigma_x$. The standard deviation is used to assess the effectiveness of smoothing noise in [72], and is calculated over a homogeneous area. A standard measure of speckle suppression is the standard deviation of the noise remaining after filtering [64]. This parameter is known as the Coefficient of Variation, CV, and can be estimated over a homogeneous or structure free region, x, in an image by,

$$CV = \frac{1}{SNR} = \frac{\sigma_x}{\overline{x}}. \tag{3.28}$$

Another traditional measure of noise smoothing is the Root Mean Squared Error, $RMSE$. This can be calculated when processing synthetically corrupted data, and the original image is known.

$$RMSE = \sqrt{\frac{1}{XY}\sum_{x=1}^{X}\sum_{y=1}^{Y}\{I(x,y)_{orig} - I(x,y)_{filt}\}^2} \tag{3.29}$$

An evaluation of speckle filters from the viewpoint of texture preservation is presented in [30] [29]. The assessment of speckle filters is carried out using the first and second-order histograms (GLCM in Section 1.3). The criteria used for a quantitative measure of speckle suppression in [29] are:

1. Preservation of the mean;
2. Reduction of the standard deviation due to noise;
3. Preservation of edges;
4. Preservation of intrinsic texture;

As stated earlier, a good speckle filter is one which reduces the speckle of an image while preserving edges. This is a difficult task because the former requires the suppression of high frequency components, while the latter requires the preservation of high frequency components. This can be described by a 1-dimensional example. Fluctuations of a signal can be due to either noise oscillations or edges, and an appropriate filter should react differently to each fluctuation. Figure 3.8 shows the ideal filter outputs for these two types of fluctuations, and implies in uniform regions the average should be calculated regardless of local oscillations, while the average should only be calculated in the peak or trough region if an edge is encountered. In general, local oscillations are higher in frequency and smaller in amplitudes when compared to signal edges.

A representative SAR image used for the evaluation of the speckle suppression algorithms is shown in Figure 3.9. Such images were selected as they possessed both

(a) Noise oscillation. (b) Signal edge.

Figure 3.8: Reactions of an appropriate filter to signal fluctuations. The signal fluctuations are shown on the left, whilst the ideal filter output is shown on the right.

uniform and edge areas. These ares are shown in Figure 3.9 and consist of 64×64 pixel blocks.

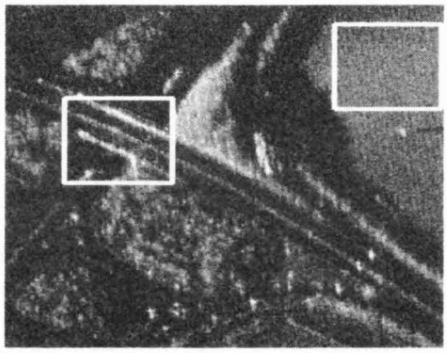


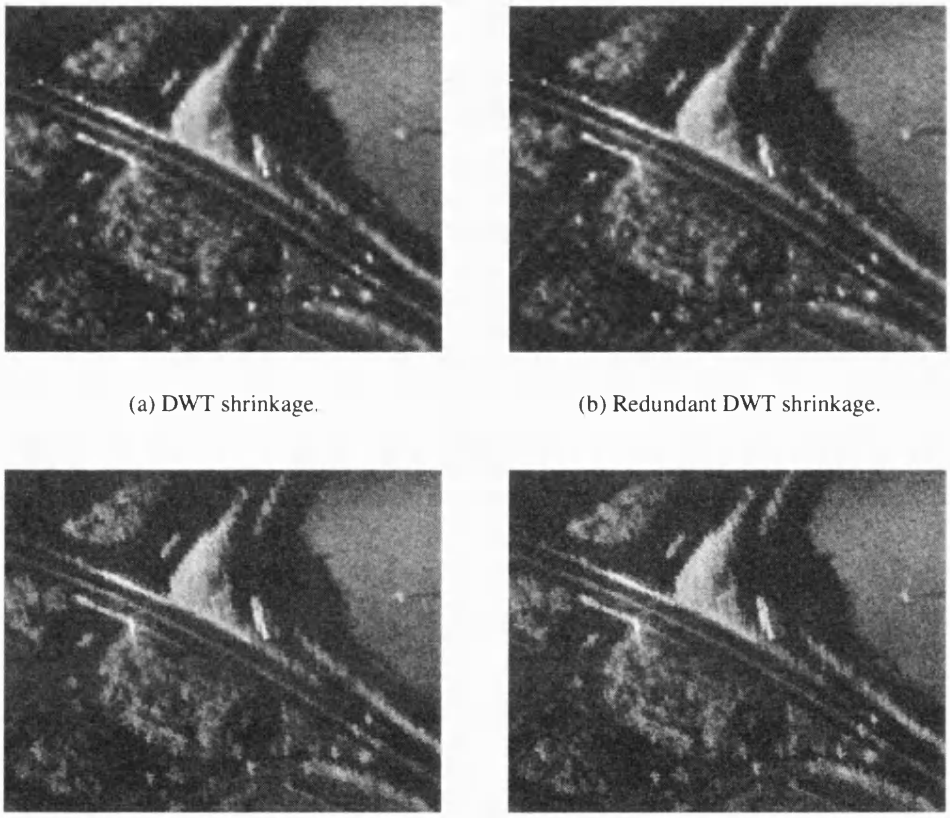
Figure 3.9: SAR image for speckle reduction.

The four algorithms:

1. Non-redundant wavelet shrinkage;
2. Redundant wavelet shrinkage;
3. Area open-close;
4. Granulometric shrinkage;

previously described, have been selected for the evaluation of speckle reduction. The wavelet transform was implemented with wavelet shrinking, Section 3.3.1, in both the redundant and non-redundant wavelet domains. Speckle reduction via mathematical morphology is achieved with area opening and closing, and granulometric shrinking.

The speckle reduced images produced by all four filtering algorithms are presented in Figure 3.10 and are evaluated using the 4 criteria of [29] in the following subsections.



(a) DWT shrinkage.

(b) Redundant DWT shrinkage.

(c) Area open-close.

(d) Granulometric shrinkage.

Figure 3.10: SAR speckle reduction.

3.5.1 Standard Deviation Reduction

A reduction of speckle noise is at the cost of losing local detail, *e.g.* a larger moving average window will lead to a higher reduction in standard deviation,in comparison to a smaller window, but will also blur more local detail. This implies a simple comparison of the reduction of standard deviation is an incomplete measure of speckle

reduction. The comparison needs to be connected with the preservation of edge information. An alternative method to assess the performance of speckle filters is to observe the preservation of edges and local detail when the reduction of standard deviation in homogeneous regions is similar for all the filters. For the results in Figure 3.10 quantitative measures for the two areas of interest are presented in Table 3.3, which shows an approximately constant reduction in standard deviation for the homogeneous area by all 4 techniques.

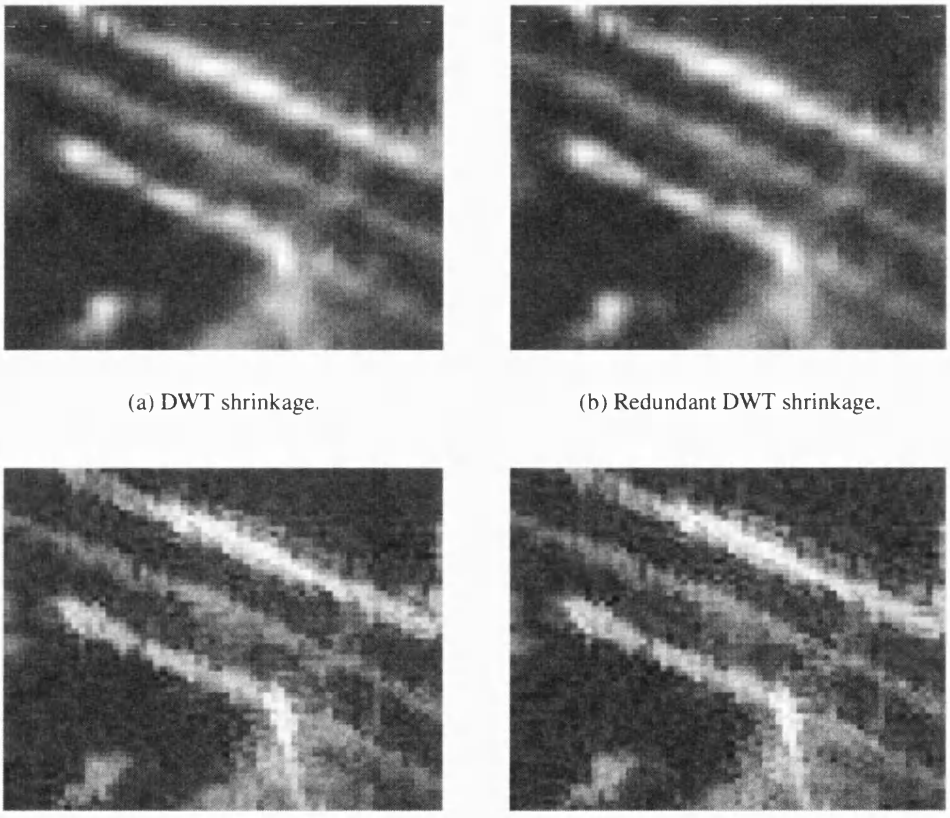
Image	Homogeneous Area			Edge Area		
	$\overline{z}$	σ_z	CV_z	$\overline{z}$	σ_z	CV_z
Original	118.8	16.787	0.1413	98.00	40.75	0.416
DWT shrinkage	117.9	7.679	0.0651	96.16	34.21	0.356
Redundant DWT shrinkage	117.9	7.678	0.0652	96.16	34.21	0.356
Area open-close	118.7	9.614	0.0810	97.08	34.38	0.354
Granulometric shrinkage	118.4	10.18	0.0859	96.49	35.12	0.364

Table 3.3: SAR speckle reduction evaluation.

3.5.2 Edge Sharpening

For homogeneous regions a reduction in standard deviation is a good measure for the efficiency of a speckle reduction filter. However for areas containing edge information an increase or decrease in standard deviation does not imply good or bad speckle reduction. This is because standard deviation is a first-order statistic and does not account for the spatial positioning of pixels, which is relevant information for image edges. Table 3.3 shows a similar value of standard deviation for all four filters at edges, however a visual inspection of Figure 3.11 shows a difference in the sharpness of the edges between the wavelet and morphology based algorithms. Analysis of Figure 3.11 indicates the speckle reduction techniques using mathematical morphology operators perform better at preserving edges. This is because morphological operators have the property of preserving shape characteristics [57], whilst the wavelet transform is implemented using linear convolutional filters.

Another way to evaluate the efficiency of speckle reduction filters is to analyse the data in the frequency domain. Each 64 $\times$ 64 pixel region of interest was treated as



(a) DWT shrinkage.

(b) Redundant DWT shrinkage.

(c) Area open-close.

(d) Granulometric shrinkage.

Figure 3.11: SAR edge area speckle reduction.

1-dimensional data consisting of 4094 samples, taken as consecutive rows of pixels in the block. The DFT is computed for each region and the difference between frequency components of the original data and filtered data is calculated. Figures 3.12 and 3.13 show the frequency analysis for the homogeneous and edge regions respectively. This difference of frequency component can then be analysed as follows:

Homogeneous region According to the principal of MMSE, in homogeneous regions filters should act as a mean filter. A positive difference in frequency component indicate a reduction in the given frequency component. If the majority of the frequency components are reduced this leads to the data being smoothed in the spatial domain.

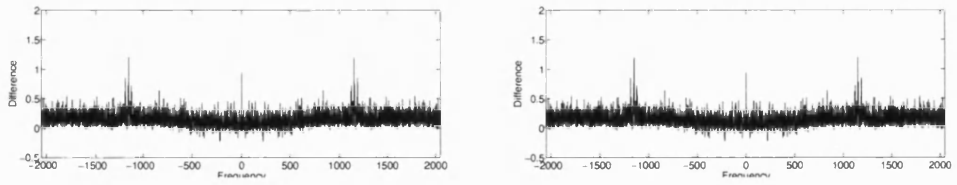
Edge region A significant amount of positive differences indicates edges will be smoothed. A small difference in frequency components means the original data has been maintained. Maintaining the original data has the advantage of preserving the edge, but has the disadvantage that little or no speckle noise is removed. Any negative differences indicate frequency components have been added resulting in edges being sharpened.

For the homogeneous region Figure 3.12 shows the majority of the differences to be positive and hence a smoothing of the image in the spatial domain for the homogeneous region. This has the result of reducing the speckle noise. Analysis of Figure 3.13 shows that in the edge region, again the majority of the frequency component differences to be positive meaning speckle has been removed, however the presence of negative differences implies frequency components have been added to sharpen the edges.

3.5.3 Texture Preservation

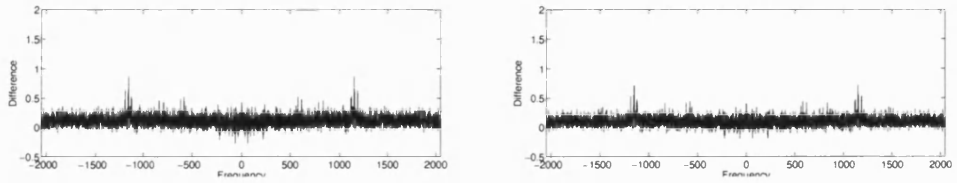
Spatial variation in a SAR image can be due to two components, fluctuations due to scene texture, and fluctuation due to speckle noise. Assuming the product model, Section 1.2.1, the intensity of a SAR image at a given pixel is given by

$$I(x,y) = \overline{I}(x,y)T(x,y)S(x,y), \tag{3.30}$$



(a) DWT shrinkage.

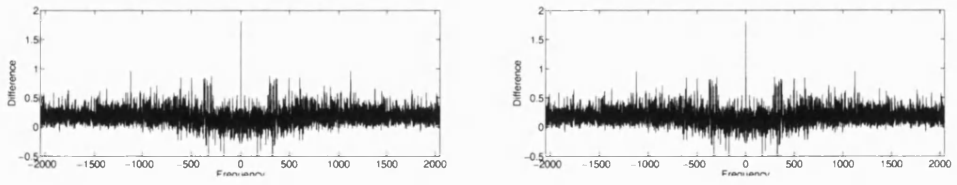
(b) Redundant DWT shrinkage.



(c) Area open-close.

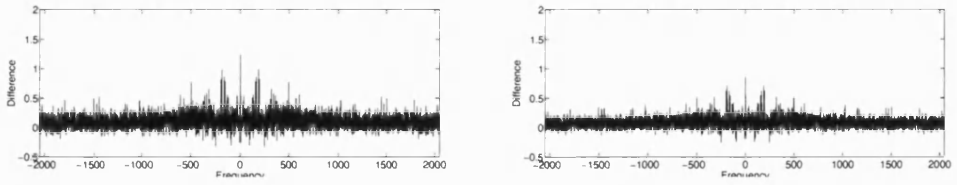
(d) Granulometric shrinkage.

Figure 3.12: Frequency component differences between original and filtered homogeneous area.



(a) DWT shrinkage.

(b) Redundant DWT shrinkage.



(c) Area open-close.

(d) Granulometric shrinkage.

Figure 3.13: Frequency component differences between original and filtered edge area.

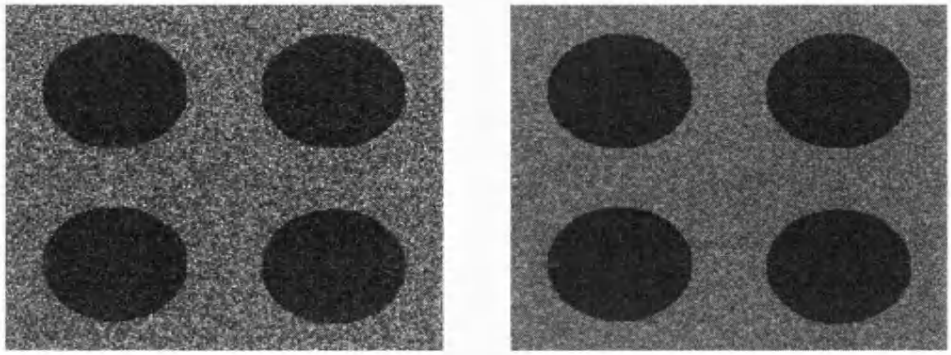
where $\overline{I}(x,y)$ is the mean value at pixel (x,y), $T(x,y)$ is the scene texture function and $S(x,y)$ is the speckle function and is independent of $T(x,y)$. For textureless regions, where $T(x,y)$ is constant, any spatial variation is due to speckle noise corruption. Although it is difficult to separate speckle texture from scene texture, it can be suppressed by speckle filters. An efficient speckle filter with respect to texture preservation should reduce speckle texture while not distorting scene texture. To evaluate the distortion of scene texture by speckle suppression two versions of an image are required. One version needs to be corrupted by a high speckle level (HSL) while the other is corrupted by low speckle level (LSL) [29]. After the HSL image is filtered, the distortion of the scene texture can be measured by comparing the filtered image with the LSL image. Here, a simulated image is used, with two distinct classes: circles (grey level = 50) and background (grey level = 100). The synthetic SAR images, Figures 3.14(a) and 3.14(b), are generated by corrupting the original image with 10 and 50 look Gamma noise (1.8) to produce the LSL and HSL images respectively.

The HSL image was filtered using the four speckle reduction algorithms described in Section 3.5 and results are given in Figure 3.14. The speckle reduction levels are recorded in Table 3.4, and show the speckle level in the filtered images are an improvement in comparison to the speckle level present in the LSL image. However, visual inspection of Figure 3.14 shows grain artefacts in the filtered images indicating the images have been distorted in the filtering process.

Image	Background			Circles		
	$\overline{z}$	σ_z	CV_z	$\overline{z}$	σ_z	CV_z
Original (HSL)	99.64	31.5	0.317	49.42	15.8	0.320
Original (LSL)	99.55	14.2	0.143	49.55	7.04	0.142
DWT shrinkage	93.95	7.25	0.0771	48.06	4.70	0.0979
Redundant DWT shrinkage	93.96	7.26	0.0772	48.05	4.71	0.0979
Area open-close	94.88	8.63	0.0909	49.18	5.76	0.117
Granulometric shrinkage	94.64	10.9	0.115	49.00	6.75	0.138

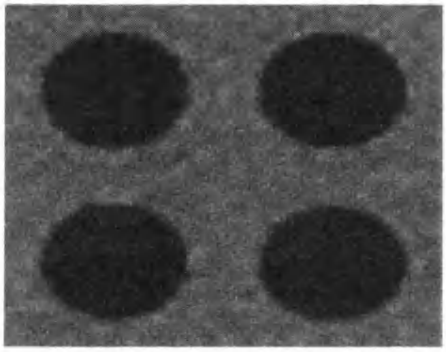
Table 3.4: Synthetic speckle reduction evaluation.

An advantage of processing images corrupted by synthetic speckle is the original data is known. Therefore as well as the speckle reduction measures in Table 3.4, the $RMSE$ (3.29) can be calculated. Table 3.5 shows the $RMSE$ between both the cor-

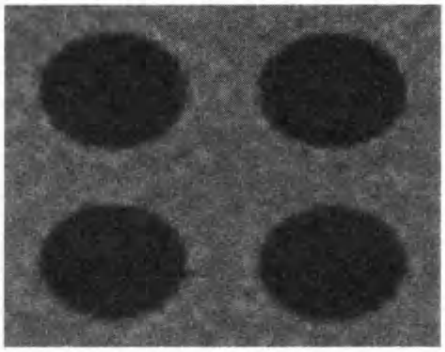


(a) Original with HSL.

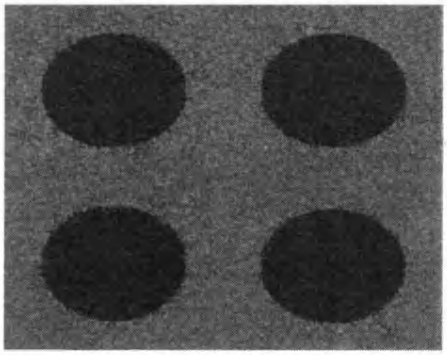
(b) Original with LSL.



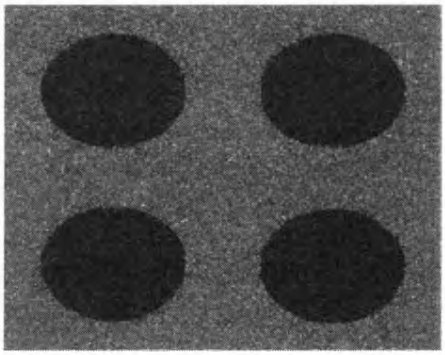
(c) DWT shrinkage.



(d) Redundant DWT shrinkage.



(e) Area open-close.



(f) Granulometric shrinkage.

Figure 3.14: Test images corrupted with synthetic speckle, shown with speckle reduction results for HSL

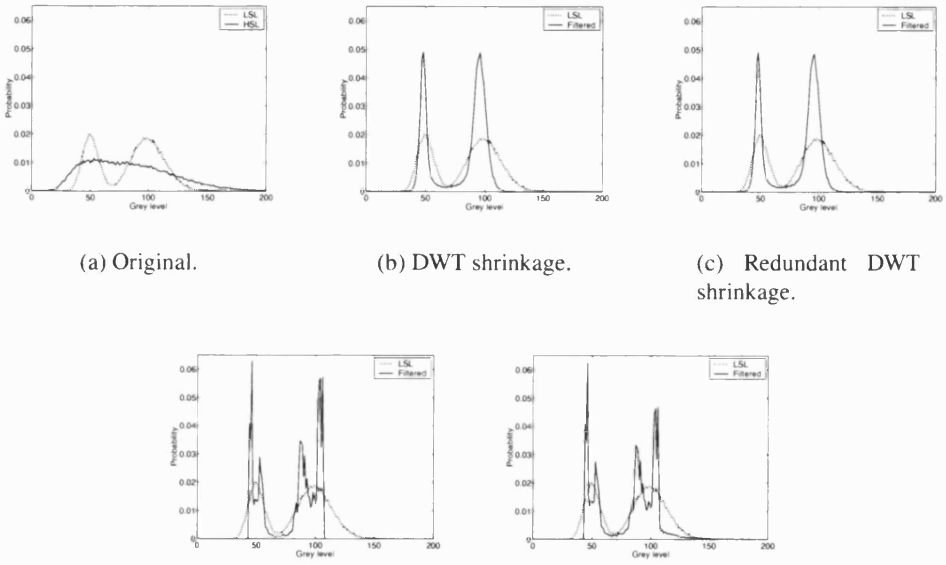
rupted and filter images, and the original image. Analysis of Table 3.5 shows the grain shrinkage filter performing slightly worse than the other filters.

Image	$RMSE_{\text{background}}$	$RMSE_{\text{circles}}$
Original (HSL)	31.551	15.846
Original (LSL)	14.230	7.056
DWT shrinkage	9.438	5.087
Redundant DWT shrinkage	9.443	5.090
Area open-close	10.031	5.813
Granulometric shrinkage	12.124	6.820

Table 3.5: Synthetic speckle reduction RMSE.

Texture preservation analysis can be achieved by viewing the histograms (Figure 3.15) and GLCM (Figure 3.16) of the filtered images. In Figure 3.15(a) the two classes for the HSL image are indistinguishable, and for the LSL case the classes are separable. The histograms from the wavelet based filters indicate the mean value of the two classes has been preserved. This is expected as the soft thresholding function, Figure 3.3(b), has been used to process the wavelet coefficients. This function modifies both positive and negative coefficients and thus preserves the mean value. In comparison the morphological based filters produce a shift in the mean value of the two classes in the low value direction. This is because the images are corrupted with Gamma noise which has an asymmetrical distribution, see Figure 1.6. The density of pixel values less than the mean is greater than the density of pixels greater than the mean. During the morphological processing the majority of the pixels modified by the area operators will be located at the higher end of the distribution. Consequently a bias toward the low value pixels is introduced.

The degradation of scene texture is not shown in the first-order histograms, Figure 3.15. An effective method of extracting textural features from an image is with the second-order histogram, or GLCM [56] [14]. The GLCM are calculated for the noisy and filtered images, with an orientation of 0° and distance of 1 pixel, and are plotted in Figure 3.16. From GLCM theory, Section 1.3, it is understood the the original speckle-free image containing the two grey values G_i and G_j, should plot on the four coordinates (G_i, G_i), (G_i, G_j), (G_j, G_i) and (G_j, G_j). The convergence to these four points can be seen when comparing the plots of the HSL image and LSL image. For the speckle reduction techniques based on the linear wavelet transform, the GLCM show a bias to-



(a) Original.

(b) DWT shrinkage.

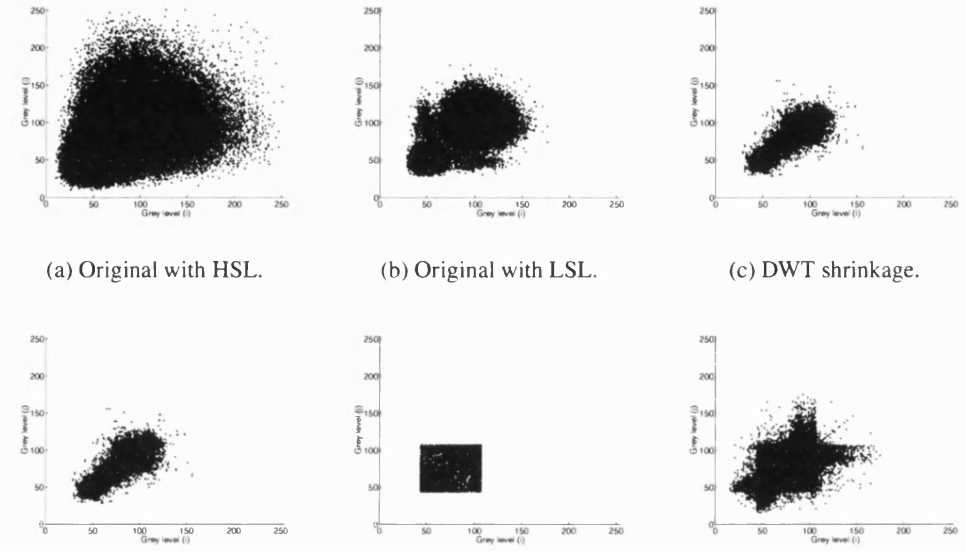
(c) Redundant DWT shrinkage.

(d) Area open-close.

(e) Granulometric shrinkage.

Figure 3.15: Synthetic speckle reduction histograms.

wards the diagonal of the matrices, indicating correlation between adjacent pixel pairs. For the morphological area open-close filter the corresponding GLCM, Figure 3.16(e), shows strong convergence the the four points, indicating improved performance in the edge regions of the image.



(a) Original with HSL.

(b) Original with LSL.

(c) DWT shrinkage.

(d) Redundant DWT shrinkage.

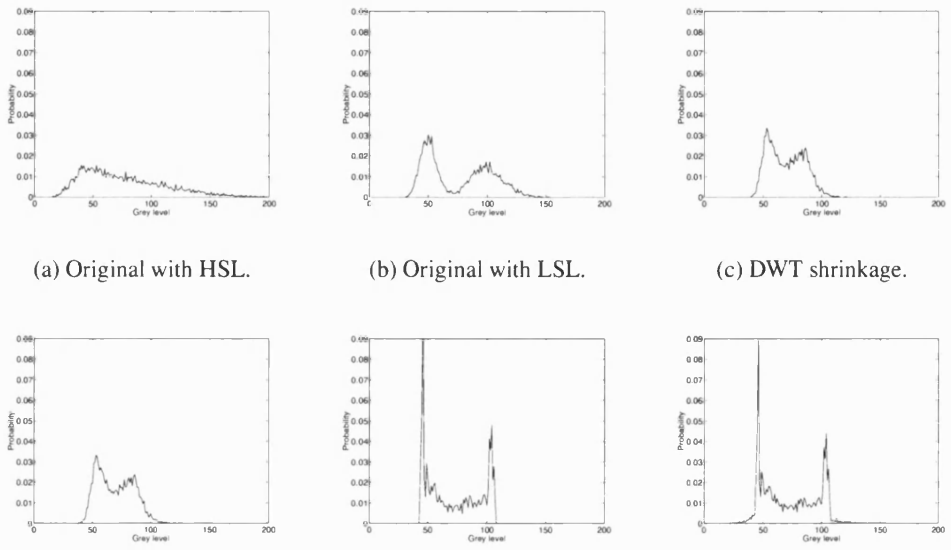
(e) Area open-close.

(f) Granulometric shrinkage.

Figure 3.16: Synthetic speckle reduction GLCM, with separation $d = 1$ pixel and orientation $\theta = 0°$.

To assess each speckle reduction techniques ability to preserve edges the data within close proximity to the edge regions of the image was examined. Since the exact positions of the edges can be calculated from the original image, a pixel is defined as being within an edge area if its distance to an edge is less than two pixels. Histograms from the data in the edge areas are plotted in Figure 3.17. Table 3.4 shows the speckle reduction for all the filters in homogeneous regions are comparable. The histograms in Figure 3.17 indicate the efficiency of the filters in edge areas. In Figure 3.15(a), the histogram for the image corrupted with HSL, the two classes are indistinct. All the filters show a separation to the classes in the edge areas, with the wavelet based algorithms providing less separation than the LSL image. For the morphology based

filters the peak for the circle class is sharper, indicating that edges have been enhanced.



(a) Original with HSL. (b) Original with LSL. (c) DWT shrinkage.

(d) Redundant DWT shrinkage. (e) Area open-close. (f) Granulometric shrinkage.

Figure 3.17: Synthetic speckle reduction edge histograms.

The test imagery in Figure 3.14, used to provide a measure of texture preservation, does not posses features reminiscent of SAR imagery. For a true evaluation of SAR speckle reduction and texture preservation a remotely-sensed speckle-free image is required. After the application of speckle reduction techniques to a synthetically corrupted version of the speckle-free image, a more accurate comparison of the texture preserving properties of the different techniques can be obtained. Figure 3.18(a) presents a SPOT image taken over Los Angeles, chosen as it contains homogeneous, edge and textured areas. A 64×64 pixel block of each of these regions is marked in black on the image. To synthesise a speckle corrupted image, the image in Figure 3.18(a) was corrupted by a 7-look correlated speckle noise model. This was achieved by using a gamma noise model to a generate single point noise image and then introducing spatial correlation by convolving with a linear filter [18]. The linear filter used was a 2-D Gaussian with $\sigma = 2/3$, resulting in correlation over an area of approximately 16 pixels. Following

the product model (1.3), the noise was then multiplied by the original image to produce the simulated image shown in figure 3.18(b).

The speckled image was processed using the wavelet and granulometric shrinkage techniques. As the presence of speckle only affects wavelet coefficients up to a scale corresponding to speckle grain size, thresholding wavelet coefficients above this level does not remove speckle but instead suppress image features [115]. Therefore the shrinkage technique was only applied to 2 levels of the wavelet decomposition, as this corresponds to the correlation size of the synthetic speckle in the test image. The threshold used was $t = \gamma\sigma_{HH} = 0.11422$, where the constant $\gamma = 2$, and the resulting speckle reduced image is shown in figure 3.19(a).

Similarly, granulometric shrinkage was applied to the residual images $G_1, G_2, \ldots, G_{16}$, using the threshold value $t = \gamma\sigma_{G1} = 0.0603$, see Figure 3.19(b). For comparison, an area open-close algorithm was applied up to an area equal to the speckle grain size, effectively removing all structures up to the size $\lambda = 16$. An area open-close up to an area size, λ, equates to the granulometric shrinkage process on the set of images $\mathbf{G} = \{G_1, G_2, \ldots, G_\lambda\}$ with the threshold t being set to the maximum value of $\mathbf{G}$.

The mean, standard deviation and CV for the homogeneous, edge and textured regions are presented in Table 3.6. All of the speckle reduction techniques preserve the mean of the homogeneous region, although it is noted that the morphological methods reduce the mean by a small percentage. However, they also show a greater reduction in the standard deviation than wavelet shrinkage, indicating an improved speckle suppression capability. A similar pattern is shown for the edge and textured regions. However, here a reduction in standard deviation does not necessarily imply improved speckle reduction as, even if the CV of the speckle-reduced images is identical to that of the original, significant edge and texture modification could have occurred.

To better assess the difference between the original and speckle suppressed noisy images for edge and textured regions the $RMSE$ (3.29) is used, see Table 3.7. For completion, results for the whole image and homogeneous region are also shown. It can be seen that for all cases the morphological-based speckle reduction algorithms out perform the wavelet shrinkage method. For the whole image and the homogeneous region the area open-close has the lowest $RMSE$, although the granulometric shrink-

age results are only slightly higher. For the edge and textured regions the situation is reversed, with granulometric shrinkage producing the lowest $RMSE$, indicating that this technique has the best speckle reduction with feature preservation.

Image	Homogeneous Region			Edge Region			Textured Region		
	$\overline{z}$	σ_z	CV_z	$\overline{z}$	σ_z	CV_z	$\overline{z}$	σ_z	CV_z
Original Image	62.9	0.93	0.02	89.4	22.49	0.25	119.4	21.97	0.18
Noisy Image	63.2	9.48	0.15	89.0	27.24	0.31	119.6	28.89	0.24
DWT shrinkage	62.9	7.02	0.11	88.3	24.29	0.28	118.7	24.67	0.21
Area open-close	62.0	5.60	0.09	86.4	21.05	0.24	116.3	18.97	0.16
Grain shrinkage	62.1	5.98	0.10	86.7	21.93	0.25	116.7	20.63	0.18

Table 3.6: 7-look SAR speckle reduction evaluation.

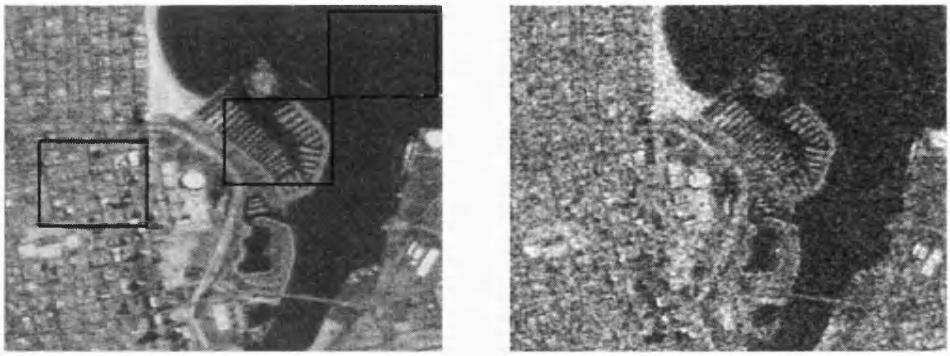
Image	$\text{RMSE}_{\text{image}}$	$\text{RMSE}_{\text{homogeneous}}$	$\text{RMSE}_{\text{edge}}$	$\text{RMSE}_{\text{textured}}$
Noisy Image	16.57	9.42	14.11	18.59
DWT shrinkage	13.55	6.98	11.84	15.59
Area open-close	12.51	5.64	11.62	15.25
Grain shrinkage	12.62	6.00	11.34	15.02

Table 3.7: 7-look Synthetic speckle reduction RMSE.

An important feature of a speckle reduction algorithm is the technique's robustness to different speckle patterns. This feature shows that a speckle reduction algorithm is not data dependent and will successfully suppress speckle in a variety of input images. To test this the speckle free SPOT image, Figure 3.18(a), was corrupted with 5 different speckle patterns, all with the same speckle statistics. Each speckle corrupted image was processed by the following speckle reduction algorithms:

1. Non-redundant wavelet shrinkage;
2. Area open-close;
3. Granulometric shrinkage;

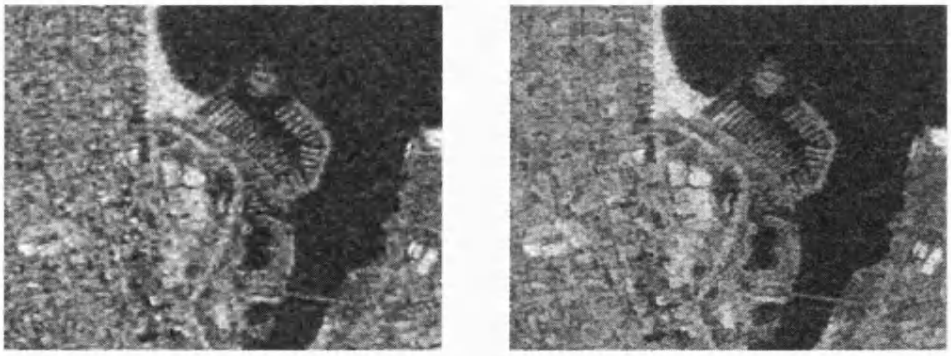
with the RMSE error calculated, for each speckle reduced image, with respect to the speckle free image. Table 3.8 shows the mean RMSE value, $\overline{\text{RMSE}}$, calculated for each of the speckle reduction techniques. To show how robust a speckle suppression technique is to different noise patterns, Table 3.8 also presents the standard deviation of the calculated RMSE error values for each technique, σ_{RMSE}. The value of σ_{RMSE}



(a) SPOT image taken over Los Angeles.

(b) SPOT image corrupted with 7-look Gamma correlated noise.

Figure 3.18: Images used for the evaluation of speckle reduction.



(a) DWT shrinkage.

(b) Granulometric shrinkage.

Figure 3.19: Results of speckle reduction techniques.

represents how the RMSE varies with different speckle patterns. It is a low value of σ_{RMSE} that implies how robust a speckle suppression technique to different speckle patterns.

Technique	$\overline{\text{RMSE}}$	σ_{RMSE}
DWT shrinkage	13.546	0.1026
Area open-close	12.407	0.0745
Granulometric shrinkage	12.501	0.0772

Table 3.8: Comparison of speckle reduction technique's robustness to different speckle patterns.

3.6 Discussion

This chapter has introduced the concept of retrieving the underlying RCS in a SAR image in the presence of speckle noise. The research has concentrated on techniques using multi-scale image analysis discussed in Chapter 2. Previous research has used speckle reduction techniques as a pre-processing stage to texture analysis [60]. Researching techniques based on multi-scale image analysis shows how speckle reduction can be incorporated in a feature extraction framework based on texture. The noise removal properties of morphological openings and closings are limited by their tendency to remove thin features along with image noise [97]. To overcome this disadvantage the novel algorithm *granulometric shrinkage* has been developed. Granulometric shrinkage has been shown to remove speckle noise whilst preserving textural information.

An evaluation of results in Tables 3.3, 3.4 and 3.6 demonstrated that all the implemented techniques exhibited good mean preservation, and provide a significant reduction in standard deviation for the homogeneous regions. Inspection of Figure 3.11, detailing on a SAR image edge region, show that the linear filterbank of the wavelet-based techniques introduce the blurring of edge features while the morphological methods, due to their property of strong causality, are able to preserve the significant edges in the region. This fact is reiterated by evaluation of the histograms in Figure 3.17, which show the morphological based techniques providing better sepa-

ration between the two images classes of Figure 3.14 in edge regions. Assessment of the preservation of texture is achieved by evaluating the $RMSE$ between a known reference image and speckle-reduced corrupted images. Table 3.5 shows speckle reduction techniques using wavelet analysis out performing morphologically based methods. However, the simple test image in Figure 3.14 does not exhibit features reminiscent of SAR imagery. A more accurate measurement of a speckle reduction technique's ability to preserve texture in SAR imagery can be seen in Table 3.7, which shows granulometric shrinkage performing best in the edge and textured regions.

The results presented in Table 3.8 show that for the speckle reduction algorithms tested the morphology based speckle reduction techniques are more robust to different speckle noise patterns in comparison to wavelet shrinkage. A robust speckle reduction technique is not data dependent, and is therefore attractive when speckle reduction is to be used as a pre-processing stage to, or integrated into, a texture analysis algorithm.

Xiao *et al.* [143] highlight the importance of a flexible filter where the ability to achieve a range of balances between the speckle reduction and detail preservation properties is desirable for different applications. Analysis of Table 3.7 shows that improved speckle reduction in the homogeneous region could be achieved for granulometric shrinkage by setting a higher value of the threshold t. This is because as t increases the granulometric shrinkage process approaches the area open-close algorithm. Therefore by using local image features to define an adaptive threshold, an optimum speckle reduction algorithm for an entire image can be achieved. Although area morphology is better at reducing speckle, granulometric shrinkage is an attractive algorithm as it keeps the smaller scale texture features. This makes granulometric shrinkage a suitable speckle reduction technique when being used a part of a morphologically based texture analysis framework.

Chapter 4

Gabor Texture Analysis

IN this chapter the use of Gabor theory for texture analysis is investigated. The construction of a Gabor filterbank has been addressed in Section 2.3, this chapter extends the theory of Gabor filters to the applicaltion of textural feature extraction in digital images. The classification of stationary textured images is addressed in Section 4.1. This provides a description on how information is presented once an image has been processed by a Gabor filterbank. Section 4.1.1 describes how features can be extracted from the various filtered images to be utilised as texture descriptors. A number of algorithms have been developed for the segmentation of textured images based on Gabor features and these are described in Section 4.2. Results of implemented segmentation schemes are presented in Section 4.3. This section includes results for the segmentation of a composite textured image made up of Brodatz textures [22]. The schemes are then extended to the texture segmentation of SAR imagery.

4.1 Gabor Texture Classification

Fully-textured images can be decomposed into a finite set of Gabor elementary functions [98]. Discrimination of fully textured images can be achieved using local features obtained via a Gabor filterbank. According to Gabor image analysis, an image $I(x, y)$

can be represented as the linear combination of Gabor functions by

$$I(x,y) = \sum_{m_x n_x m_y n_y} a_{m_x n_x m_y n_y} \cdot f_{m_x n_x m_y n_y}(x,y), \tag{4.1}$$

where $f_{m_x n_x m_y n_y}$ is a Gabor function of the order $(m_x n_x m_y n_y)$ and

$$f_{m_x n_x m_y n_y}(x,y) = g(x - m_x D_x, y - m_y D_y) \cdot \exp j(n_x W_x x + n_y W_y y), \tag{4.2}$$

where $g(x,y)$ is a normalised two-dimensional window. The function $f_{m_x n_x m_y n_y}$ is located at the point $(x = m_x D_x, y = m_y D_y)$ and has spatial frequency components $(\omega_x = n_x W_x, \omega_y = n_y W_y)$. The constants D_x, D_y and W_x, W_y are the sampling intervals along the spatial and spatial-frequency axes respectively. In [98], equal sampling intervals are defined along the x and y axes in both the spatial and spatial-frequency domains, *i.e.* $D_x = D_y = D$ and $W_x = W_y = W$. The coefficient $a_{m_x n_x m_y n_y}$ is of the order $(m_x n_x m_y n_y)$, and represents the relative weight of the corresponding Gabor function in the image $I(x,y)$. In theory an infinite number of Gabor coefficients are required to represent an image. However, Porat [98] demonstrates how a relatively small number of coefficients produces a good reconstruction.

4.1.1 Gabor Textural Feature Extraction

With the assumption that the fundamental primitives of natural textures are localised frequency components in the form of Gabor functions, textures can be analysed by the correlation of such primitives with natural textures. Along with positional information defined along the Gabor lattice, features describing image texture can be calculated relating to the Gabor parameters. The feature extraction is based on three basic features:

1. spatial-frequency along the preferred orientation
2. orientation of spatial-frequency
3. image intensity

For each of the three basic features, the first and second moments can be calculated resulting a 6-dimension feature space.

Prior to feature extraction, an image is represented by the set of Gabor coefficients $\{a_{m_x n_x m_y n_y}\}$, where a coefficient with indexes m_x and m_y represents the spectral components within a local neighbourhood centred on the pixel $x = m_x D_x, y = m_y D_y$. The relative spatial-frequencies are defined by the harmonic numbers n_x and n_y, such that $\omega_x = n_x W_x$ and $\omega_y = n_y W_y$. When an image is expressed in this manner the absolute value of the coefficient specified by indexes n_x and n_y is proportional to the power of the spatial-frequency components (ω_x, ω_y) over the area defined by (m_x, m_y) and D. The frequency components (ω_x, ω_y) can conveniently be expressed as polar coordinates, detailing spatial-frequency, $\omega = \sqrt{\omega_x^2 + \omega_y^2}$ and orientation, θ, where $\tan(\theta) = \omega_y/\omega_x$. As orientation is the most noticeable parameter characterising the receptive field of cortical cells, it can be assumed that orientation is fundamental to the description of texture. The first feature of the feature space is the dominant localised frequency, F, over the area defined by m_x, m_y and calculated by

$$F_{m_x m_y} = \frac{\sum_{n_x=1}^{N-1} \sum_{n_y=1}^{N-1} |a_{m_x n_x m_y n_y}| \sqrt{n_x^2 + n_y^2}}{\sum_{n_x=1}^{N-1} \sum_{n_y=1}^{N-1} |a_{m_x n_x m_y n_y}|}, \tag{4.3}$$

where N is the number of spectral components determined by the sampling rate of the image. The term $\sqrt{n_x^2 + n_y^2}$ denotes the harmonic number ω with respect to n_x and n_y. As the spatial frequencies are included regardless of orientation, the feature F is rotation invariant. The variance of the dominant localised frequency, VF, is a scale invariant feature calculated by

$$VF_{m_x m_y} = \frac{\sum_{n_x=1}^{N-1} \sum_{n_y=1}^{N-1} |\sqrt{n_x^2 + n_y^2} - F_{m_x m_y}|}{N^2}. \tag{4.4}$$

This feature provides information regarding the bandwidth of the localised frequency and defines the regularity of a texture. Two more features are defined that are dependent on orientation of the spatial frequencies. The orientation of a Gabor function in terms of n_x and n_y is $\tan[\theta(n_x, n_y)] = (n_y/n_x)$ for $n_x \neq 0$. The dominant local

orientation, T, is calculated in a similar manner as (4.3) by

$$T_{m_x m_y} = \frac{\sum_{n_x=1}^{N-1}\sum_{n_y=1}^{N-1} |a_{m_x n_x m_y n_y}|\theta(n_x, n_y)}{\sum_{n_x=1}^{N-1}\sum_{n_y=1}^{N-1} |a_{m_x n_x m_y n_y}|}. \tag{4.5}$$

The feature based on local orientation is, as expected, sensitive to rotation and are useful in detecting edges caused by the rotation of the same texture. The second moment of the local orientation, VT, is given by

$$VT_{m_x m_y} = \frac{\sum_{n_x=1}^{N-1}\sum_{n_y=1}^{N-1} |\theta(n_x, n_y) - T_{m_x m_y}|}{N^2}. \tag{4.6}$$

The local mean intensity, $Lm_x m_y$, is calculated to characterise smooth regions of texture where the first four features do not provide sufficient discrimination. This is because smooth regions of an image contain little spatial frequency information, and require local intensity information to provide separation in the feature space. A local measure of the mean intensity is achieved by multiplying the intensity image $I(x, y)$ with the window function $g(x, y)$. The local mean intensity, $Lm_x m_y$, is the average pixel value over the resulting area such that

$$L_{m_x m_y} = \frac{1}{K} \sum_{x,y \in A(m_x m_y)} I(x, y), \tag{4.7}$$

where $A(m_x m_y)$ is the set of pixels belonging to the area defined by the window function $g(x, y)$ centred at m_x, m_y. The variance of the local intensity, VL, is calculated by

$$VL_{m_x m_y} = \frac{1}{K} \sum_{x,y \in A(m_x m_y)} |I(x, y) - L_{m_x m_y}|, \tag{4.8}$$

and provides discrimination that are similar according to the first five features, and is useful in the classification of irregular textures.

The features (4.7) and (4.8) are in no way related to the spatial frequency. This can be demonstrated by the fact that the pixels defined by the area $A(m_x m_y)$ can be positioned at different spatial locations, without affecting the mean or variance of the local inten-

sity. The six features produce the feature vector$\{F, VF, T, VT, L, VL\}_{m_x m_y}$ which is position-dependent and is parameterised by m_x, m_y. Therefore, for every localised area defined by m_x, m_y a six-dimensional feature space is define which can be applied to a classification scheme.

Porat and Zeevi show that the full six-dimension feature space is not necessary for classification and a specified sub-set of the feature space can still provide accurate classification [98]. When classifying a set consisting of eight reference textures, taken from [22], the single feature VF (4.4) provided 62.5% classification accuracy. The most informative textures are VF (4.4) and VL (4.8) even in the presence of Gaussian noise. The eight natural textures from [22] are shown to form clusters when plotted in the feature space VF verses VL. Using this two-dimensional feature space the textures were classified using a supervised minimum distance classifier, which was demonstrated to be invariant of translation and rotation, whilst providing accurate classification results in the presence of Gaussian noise.

4.2 Gabor Texture Segmentation

Gabor functions (2.7) are suitable for the local image texture analysis, as they have the fundamental property of being localised. Jain and Farrokhnia present a texture segmentation algorithm based on multi-channel filtering [62]. The channels were characterised by a bank of Gabor filters that provided almost uniform coverage in the spatial-frequency domain, see Figure 2.6. The Gabor filterbank is constructed using the radial frequencies and 4 orientation values described in Section 2.3. The filters with the 2 lowest values of radial frequencies are excluded from the filterbank as they capture spatial variations that are too large to describe textural information.

Using only a subset of filtered images can reduce the computational burden as this results in a reduction of texture features. Let $s(x, y)$ be the reconstruction of the original image by adding all the pixel responses of the filtered images, and $\hat{s}(x, y)$ be the *partial* reconstruction of the image by adding a subset of filtered images. The sum of

squared error, SSE, introduced by using $\hat{s}(x,y)$ instead of $s(x,y)$ can be measured by

$$SSE = \sum_x \sum_y [\hat{s}(x,y) - s(x,y)]^2 . \tag{4.9}$$

The fraction of intensity variations in $s(x,y)$ that is contained within $\hat{s}(x,y)$ can be measured by

$$R^2 = 1 - \frac{SSE}{SSTOT}, \tag{4.10}$$

where R^2 is the coefficient of determination, and

$$SSTOT = \sum_x \sum_y s(x,y)^2. \tag{4.11}$$

The best subset of features/subbands can be determined by the following sequential forward selection procedure:

1. Select the subband that best represents $s(x,y)$, *i.e.* gives the highest value of R^2.
2. With the previously selected subband(s), select the next subband which best represents $s(x,y)$.
3. Repeat Step 2 until $R^2 \geq K$, where K is a constant less than 1.

A suitable value for K is 0.95, which means that the subset of subbands used accounts for at least 95% of the intensity variations in $s(x,y)$.

An important goal of texture analysis is to obtain a set of texture features that can be used to discriminate between arbitrary textures. Firstly, to compute texture features each filtered image is subjected to a nonlinear transform. The non-linear transform used in [62] is

$$\psi(t) = \tanh(\alpha t) = \frac{1 - \exp\{-2\alpha t\}}{1 + \exp\{-2\alpha t\}}, \tag{4.12}$$

where α is a constant. A value of $\alpha = 0.25$ was chosen as this results in a rapidly saturating, threshold-like transformation.

Instead of identifying individual regions of texture, it is desirable to calculate texture energy, which is the absolute average deviation from the mean in small overlapping

windows. Each feature image $e_k(x, y)$ is given by

$$e_k(x, y) = \frac{1}{M^2} \sum_{(a,b) \in W_{xy}} |\psi(r_k(a, b))|, \tag{4.13}$$

where $r_k(x.y)$ is the corresponding filtered image, $\psi(\cdot)$ is the non-linear function of (4.12) and W_{xy} is an MxM window centred on pixel (x, y). An important parameter in (4.13) is the size of the averaging window M. More reliable texture features are obtained with larger windows, while more accurate localisation of texture boundaries require smaller windows. It was found that Gaussian weighted windows resulted in more accurate localisation of texture boundaries, which is consistent with [141]. Weldon also states the presence of a Gaussian postfilter can reduce misclassifications due to the presence of noise in the image. In [141], the non-linearity of (4.12) is replaced by the magnitude operator.

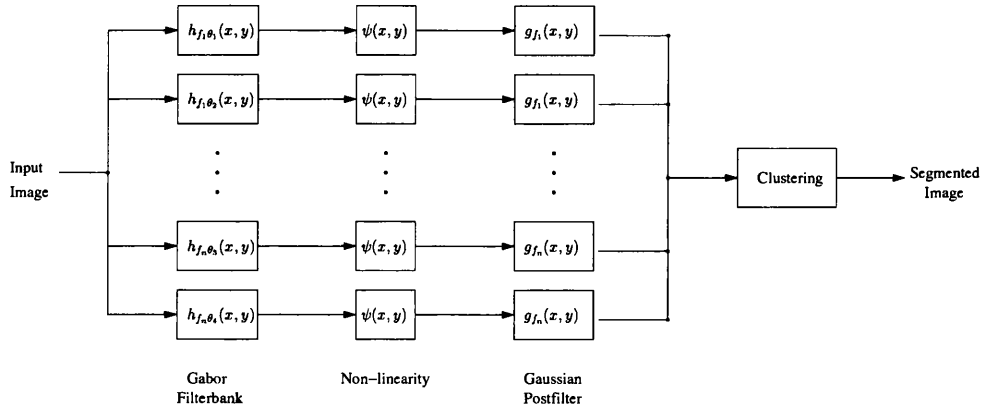


Figure 4.1: Gabor filterbank image segmentation process [62].

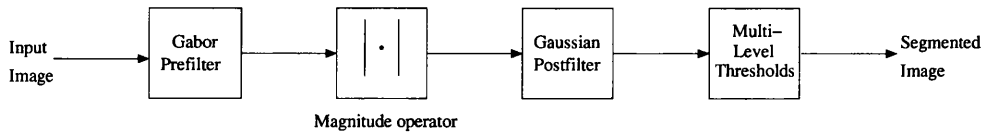


Figure 4.2: Simplified Gabor image segmentation process [141].

Once the feature images have been calculated the features corresponding to different filters need to be integrated to produce a segmentation. If the features are capable of

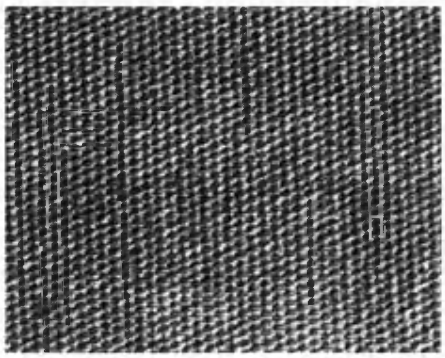
discriminating against different textures in the input image then clusters should form in the feature space. For segmentation each cluster should be compact and isolated from clusters corresponding to different textures.

4.3 Results

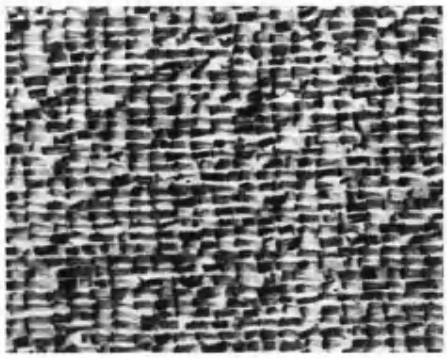
To evaluate the effectiveness of the texture segmentation algorithms a set of results are generated using a test image made up of composite textures taken from Brodatz's album [22]. The composite test image, Figure 4.3(f), is chosen to avoid straight horizontal and vertical texture edges, and is used to evaluate texture segmentation algorithms in [103]. Using a multi-scale image transform with sub-sampling for feature extraction, such as the DWT (see Section 2.4.1), this can lead to blocking artefacts in the final segmentation. If the boundaries of the blocks coincide with the texture boundaries, which is unlikely to occur in SAR images of the real world, the segmentation with the sub-sampling will have an advantage over techniques based on non-sub-sampled image transforms. Therefore the test image has been constructed with no horizontal and vertical texture boundaries.

The Gabor segmentation scheme implemented used the filterbank described in [62], giving the spatial frequencies $f_0 = \{4\sqrt{2}, 8\sqrt{2}, \ldots, (N_c/4)\sqrt{2}\}$ cycles/image width. For each Gabor function (2.7) the value, $\phi = 0$ was chosen, as this gave a symmetrical filtering function, thus preventing a spatial shift in the feature space. The corresponding dimensions of the Gaussian envelope for each spatial frequency are given by (2.8). As opposed to the sub-set of features described in Section 4.2, the full Gabor decomposition was utilised as computational burden was not seen to be a critical factor for textural feature extraction. After the Gabor filtering process the magnitude non-linear operator was applied to the filtered images [141]. A Gaussian weighted postfilter was applied to the feature images after the non-linear operator. Following [19], the spatial dimension of the Gaussian postfilter σ_{post} was set proportional to the spatial dimension of the Gabor function by

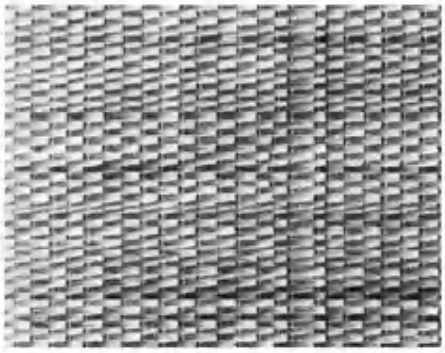
$$\sigma_{post_x} = \frac{3\sigma_x}{2} \qquad \text{and} \qquad \sigma_{post_y} = \frac{3\sigma_y}{2}, \tag{4.14}$$



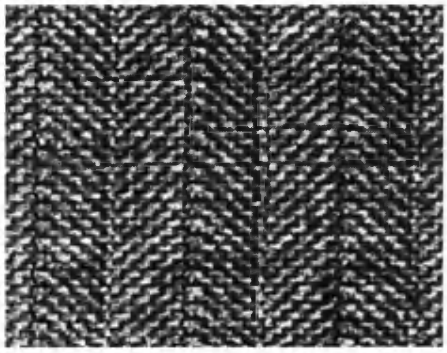
(a) Texture D77.



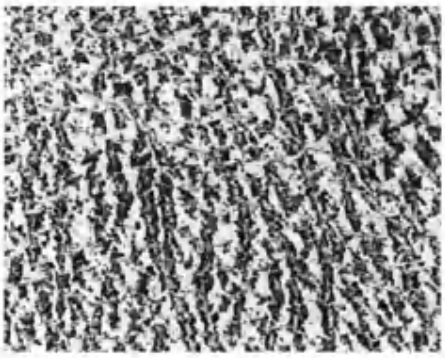
(b) Texture D84.



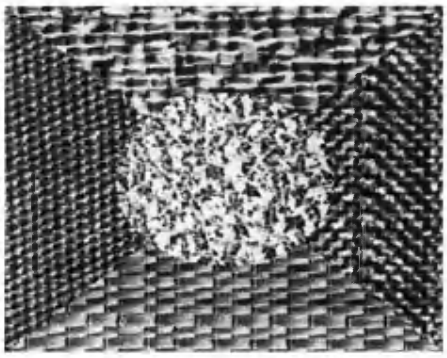
(c) Texture D55.



(d) Texture D17.



(e) Texture D24.



(f) Test image.

Figure 4.3: Brodatz textures and composite test image for segmentation.

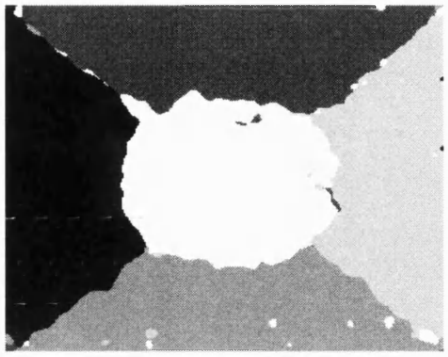


Figure 4.4: Supervised segmentation of compound Brodatz texture image utilising features extracted via a Gabor filterbank.

where σ_{post_x} and σ_{post_y} are the spatial dimension of the Gaussian postfilter and the x and y directions, and σ_x and σ_y are the spatial dimension of the Gabor pre-filter. By including the information of features within a local neighbourhood, the centre feature is adjusted so that all features within the neighbourhood will have similar membership values to a cluster centre [118]. Once a Gabor textural feature extraction process had been defined, a segmentation of the test image Figure 4.3(f) was achieved with a minimum distance classifier. In supervised texture classification schemes a feature vector, consisting of a number of textural features, is evaluated against a selected library of feature vectors for particular textures, see Section 1.3.2. The library of reference texture features was obtained by calculating the mean vector μ_k (1.9) and covariance matrix C_k (1.10), using the training images presented in Figures 4.3(a)–(e). The Mahalanobis distance, see Table 1.2, is calculated between the feature vector at every pixel position and every entry in the database of codevectors. The texture class with

True class:	Classified as:				
	D77	D84	D55	D17	D24
D77	98.09%	0.58%	0.47%	0.01%	0.85%
D84	0.23%	97.76%	0.16%	0.06%	1.79%
D55	3.02%	0.12%	93.17%	0.76%	2.93%
D17	0.08%	3.05%	1.00%	92.88%	2.99%
D24	1.11%	2.44%	0.31%	0.45%	95.69%

Table 4.1: Confusion matrix for supervised segmentation of compound Brodatz texture image utilising features extracted via a Gabor filterbank.

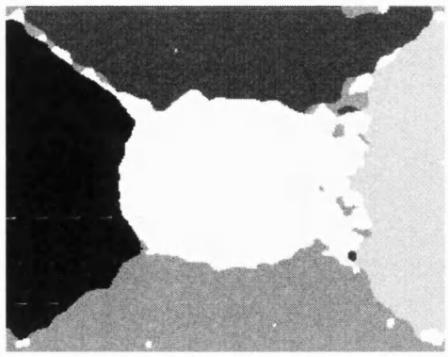


Figure 4.5: Unsupervised segmentation of compound Brodatz texture image utilising features extracted via a Gabor filterbank.

the minimum distance is then assigned to the pixel position. The segmentation result of the minimum distance classifier is presented in Figure 4.4.

A supervised segmentation algorithm, see Section 1.3.2, is dependent on the availability of suitable training data. Provided the extracted texture features allow good separation of the texture classes in the feature space, a suitable clustering algorithm can be used to achieve an unsupervised segmentation, see Section 1.3.3. The resulting segmentation obtained via the k-means clustering, see Section 1.3.3.1, of the Gabor texture features is shown in Figure 4.5.

In texture segmentation, adjacent pixels are highly probable to belong to the same texture class and the incorporation of spatial information in the feature space has been shown to reduce misclassifications in regions of homogeneous texture and at texture

True	Classified as:				
class:	D77	D84	D55	D17	D24
D77	95.54%	0.05%	1.54%	0.00%	2.87%
D84	0.00%	94.58%	2.08%	0.00%	3.33%
D55	1.59%	0.00%	95.73%	0.19%	2.49%
D17	0.00%	2.25%	3.01%	70.73%	24.01%
D24	0.47%	1.05%	0.09%	0.00%	98.38%

Table 4.2: Confusion matrix for unsupervised segmentation of compound Brodatz texture image utilising features extracted via a Gabor filterbank.

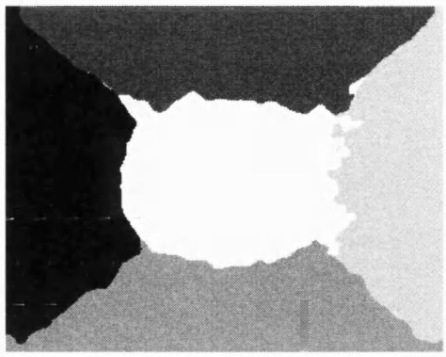


Figure 4.6: Unsupervised segmentation of compound Brodatz texture image utilising features extracted via a Gabor filterbank with spatial coordinates.

boundaries [62]. The inclusion of the spatial co-ordinates in the feature vector will reduce the distance between neighbouring feature vectors encouraging neighbouring pixels to cluster together, thus avoiding over-fragmentation of an otherwise homogeneous texture region. In practice, this can be achieved by including the spatial co-ordinates of each pixel as two extra features. As no a priori information about the co-ordinates of each texture class is available this approach is incompatible with a supervised segmentation scheme. The result achieved by this approach is presented in Figure 4.6.

Results presented in Figures 4.4–4.6 show the Gabor filterbank approach to the segmentation of the Brodatz compound textured image to be a success. The majority of misclassifications occur at the texture boundaries, due to the linear property of the filterbank. The linear convolution of the Gabor functions smooths the texture boundaries

True	Classified as:				
class:	D77	D84	D55	D17	D24
D77	97.90%	0.67%	0.56%	0.00%	0.86%
D84	0.31%	97.91%	0.00%	0.16%	1.62%
D55	1.70%	0.00%	96.89%	0.32%	1.09%
D17	0.00%	3.06%	2.25%	85.56%	9.14%
D24	0.97%	1.41%	0.40%	0.18%	97.03%

Table 4.3: Confusion matrix for unsupervised segmentation of compound Brodatz texture image utilising features extracted via a Gabor filterbank with spatial coordinates.

in the feature space, resulting in artificial regions of texture. The inclusion of spatial coordinates achieves a reduction of misclassified pixels. Inspection of the confusion matrix in Tigure 4.3 shows that intra-region classification errors are virtually eliminated with the only misclassifications occurring at region boundaries. This provides an improvement in overall correct classification from 90.99% in Table 4.2 to 95.06%. Once the Gabor texture segmentation algorithm had been verified on the composite texture image, Figure 4.3(f), the technique could be applied to SAR data. The test image shown in Figure 4.7 is an X-band SAR image taken from an airborne platform with spatial resolution in the x and y directions being 0.75m and 0.60m respectively.

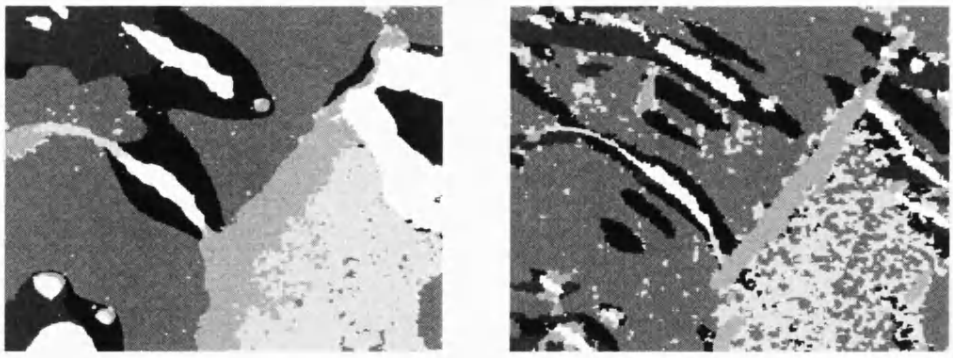
Figure 4.7: X-band SAR image for texture segmentation, consisting of 605×605 pixels.

The image in Figure 4.7 has previously been segmented using the Merge Using Moments, MUM algorithm [95]. Initially the image is massively over segmented, then a comparison of neighbouring regions is made, which leads to a list of potential merges. The list is sorted to identify the most eligible regions to be merged. The merge process is iterated until no merges become eligible, resulting in the final segmentation. The regions in the final segmented image can be described in terms of a number of features derived from 1) size; 2) shape; 3) mean intensity and 4) textural property. A suitable textural property is the coefficient of variation (2.25), which can be calculated for each region in the segmentation [93]. The classification of the regions resulted in six texture classes being present in the image [7] and this was the number of class centres used for the unsupervised segmentation of Figure 4.7, obtained via the k-means clustering of the Gabor texture features. The resulting segmented image is shown in Figure 4.8(a).

To obtain a more robust segmentation of SAR imagery, the image can be segmented as a pre-processing stage [6]. The classification of regions resulting from the segmentation stage has the effect of making the classification less sensitive to the presence of a speckle noise component. As an alternative to using Gaussian weighted windows to adjust a feature value to suit its neighbourhood, watershed segmentations can be used to define the local neighbourhood, thus reducing the degradation of edge information in the feature space. To make the segmentation consistent with the increasing size of the Gaussian postfilter dependent on the scale, the watershed transform using markers was implemented, see Section 2.5.3. Once a region has been defined, all features belonging to the region are replace by the mean value of the member features. Selecting a suitable threshold t_{ws} for the watershed markers, identifying regions of increasing area, whilst preventing the suppression of smaller regions of homogeneous texture. A threshold value t_{ws} can be calculated using

$$t_{ws} = \left(\frac{f_0}{2\sqrt{2}}\right)^2, \tag{4.15}$$

where f_0 is the frequency of the sinusoid in the Gabor prefilter and produces regions with areas that were comparable to the appropriate Gaussian postfilter dimensions. The segmentation obtained by replacing the Gaussian postfilter with segmentation regions is shown in Figure 4.8(b)



(a) Gaussian postfilter.

(b) Post watershed feature adjustment.

Figure 4.8: Gabor unsupervised texture segmentation results.

4.4 Discussion

In this chapter a texture segmentation scheme based on Gabor filters has been discussed, developed and tested. The feature extraction is achieved by multi-scale image analysis by varying the bandwidth of the Gabor function (2.7). The described segmentation algorithm is analogous with the model of the HVS [78]. After multi-scale analysis the filtered images are processed with a non-linear operator, producing feature images with improved discrimination of textures with similar single point statistics. Prior to supervised classification, or unsupervised clustering, the feature images are modified with a Gaussian postfilter, resulting in the suppression of weak features, which can lead to misclassified pixels.

Although no ground truth information is available for the image in Figure 4.7, a visual inspection of the segmentation results, Figure 4.8, shows the segmented regions to have a strong bias toward the orientation parameter θ. Limited success in the segmentation results of Figure 4.8 may be because Gabor functions capture textural features that are too coarse for the analysis of high resolution SAR imagery.

Chapter 5

Wavelet Texture Analysis

The wavelet transform described in Section 2.4.1 provides a multi-resolution decomposition of an image and results in a non-redundant image representation. This characterisation of the wavelet transform allows the study of an image from the coarse resolution to the fine resolution and the extraction of information in any of the levels of decomposition. The traditional pyramid-type wavelet transform proposed by Mallat [79] recursively decomposes subsignals in the low frequency channels. This transform is suitable for signals consisting primarily of smooth components so the majority of the information is concentrated in the low frequency regions. Since the most significant information of a texture often appears in the middle frequency channels [24], further decomposition of the low frequency region, as implemented by the wavelet pyramid transform, may not provide useful information for classification based on texture.

In this chapter the classification of stationary textured images is described in Section 5.1. Particular attention is drawn to the tree-structured wavelet transform, in Section 5.1.1, which extracts textural features from the mid-frequency regions of an image. The methodology for the classification of whole textured images is extended to look at the problem of wavelet based texture segmentation in Section 5.2. By calculating texture features locally within the wavelet coefficients (which by their nature are localised) a texture feature vector can be calculated on a pixelwise basis, providing a method for single pixel classification and hence image segmentation. The wavelet-based texture segmentation of SAR imagery [27], is discussed in Section 5.2.1. From

investigations into wavelet texture analysis, a segmentation algorithm utilising the wavelet packet transform has been developed [39], detailed in Section 5.3. From research into SAR speckle noise reduction in Chapter 3, a method for incorporating speckle reduction into the texture feature extraction framework is described in Section 5.3.1. Segmentation results for both the redundant wavelet transform and non-redundant wavelet transform are presented in Section 5.4.

5.1 Wavelet Texture Classification

Laine [69] introduced a new multiscale approach to characterise textures. A full wavelet packet decomposition was calculated for a textured image, where each packet (channel) reflects a specific scale and orientation sensitivity. It was reported that the energy associated with each packet can provide unique information about a texture and make available a representation or *signature* for classification. This signature is a texture feature vector consisting of a set of energy values. An alternative information measure is entropy, defined by

$$H(x) = -\sum_{k} |x_k|^2 \log |x_k|^2, \tag{5.1}$$

where x_k is the energy value in wavelet packet k and was proposed as a texture measure in [56].

Once a texture has been represented in feature vector form, it can be assigned to a texture class using a minimum distance classifier. A minimum distance classifier is based on the assumption that each texture class, k, can be represented by the prototype pattern μ_k, known as the class centre. A minimum distance classifier assigns an input texture pattern $\overline{x}$, of unknown classification to the texture class k if the distance D_k between $\overline{x}$ and μ_k was minimum across all possible texture classes. The Euclidean distance, see Table 1.2, was used as the distance function in [69]. Each class centre was estimated by using the mean of a training sample for each texture class

$$\mu_k = \frac{1}{N_k} \sum_{\overline{x} \in k} \overline{x}. \tag{5.2}$$

Using the estimates of the true class centres makes this algorithm a *supervised* minimum distance classifier.

For each textured input a complete set of wavelet packets were computed. A complete five level decomposition (level 0 - 4) consisted of 341 wavelet packets, see Table 5.1. The energy and entropy of each wavelet packet were calculated giving the maximum

Decomposition level	Number of wavelet packets
Level 0	1
Level 1	4
Level 2	16
Level 3	64
Level 4	256

Table 5.1: Number of wavelet packets at various decomposition levels.

signature of feature vector length of 341 elements. Instead of using a complete wavelet decomposition the classification performance using a subset of wavelet packets was investigated. The six subsets of features chosen for investigation in [69] are:

1. Complete set of wavelet packets. Each texture is represented by a 341 element feature vector.
2. Wavelet pyramid subbands. Nodes of the wavelet decomposition [79], the four low frequency nodes at each level, see Figure 2.11(a). Each texture is represented by a 17 element feature vector.
3. Levels 1, 2 and 3. Each texture is represented by an 84 element feature vector.
4. Levels 2 and 3. Each texture is represented by an 80 element feature vector.
5. Level 3. Each texture is represented by a 64 element feature vector.
6. Level 4. The 256 energy feature taken from the coarsest scale.

Table 5.1 shows that with an increase in decomposition level of the wavelet transform, the number of frequency subbands increases exponentially, whereas the size of the sub-image decreases. This is important to remember when implementing the wavelet

transform to achieve texture segmentation, see Section 5.2, as the decrease in image resolution makes the assignment of a pixel to a texture class a more difficult task [63].

5.1.1 Tree-Structured Wavelet Transform

Chang and Kuo [24] stated that low frequency regions of textured images may not contain significant information. An appropriate way to implement the wavelet transform for textures is to detect the significant frequency bands and then decompose them further. The method described is called the tree-structured wavelet transform. Figure 5.1 shows how the output from the 2 channel analysis filter bank, Figure 2.10, can be used to construct the tree-structured wavelet transform. As textural information is only present in specific frequency regions there must be some criterion to select which subbands to decompose. This is described in the tree-structured wavelet transform algorithm.

1. Decompose a given image with the two dimensional discrete wavelet transform, Figure 2.10, into four sub-images.
2. Calculate the energy of each sub-image. For a sub-image $x(m,n)$, with $1 \leq m \leq M$ and $1 \leq n \leq N$ the energy is

$$e = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} |x(m,n)|. \tag{5.3}$$

3. If the energy of a sub-image is significantly smaller then the others, it contains less information and the decomposition of the sub-image is stopped. This is achieved by comparing the sub-image energy with the maximum energy level at the same scale. That is, if $e < Ke_{max}$, stop decomposing the sub-image where K is a constant less than 1.
4. If the sub-image contains significant energy, if $e \geq Ke_{max}$, the sub-image is decomposed.

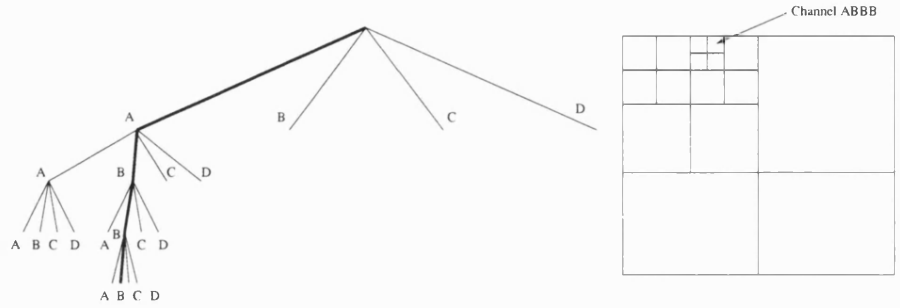


Figure 5.1: Tree-structured wavelet transform.

With respect to each tree-structured wavelet transform, the energy values at each node was calculated to obtain an energy map in the spatial frequency domain. This energy map could then be used for texture classification. The robustness of the tree-structured wavelet transform was tested by applying the transform to 100 samples of the same texture. It was discovered that for the same texture 16 different tree structures were observed. This is because the number of tree structures depends highly on the constant K. This suggests that the structure of the tree might not be the ideal candidate to use for texture representation. Instead the focus was put on identifying the dominant frequency channels with large energy values. For 100 samples of the same texture the first 5 dominant frequency channels were detected and were shown to be a more robust texture descriptor. This was the texture feature vector used in a minimum distance classifier. The codevector was generated by taking m samples of the same texture, and decomposing each with the tree-structured wavelet transform. This defines the energy map in the spatial frequency domain. The codevector μ_k is then calculated by averaging all energy maps over m samples.

Classification of an unknown texture is achieved by decomposing the texture with the tree-structured wavelet transform to construct the energy map. The energy from the first J dominant channels are chosen as the texture feature vector $\overline{x}$. For every texture in the database the energy values in the same channels are used to form the codevector. The distance of the feature vector is calculated for every texture in the database $D_k = \text{distance}\ (\overline{x}, \mu_k)$. The unknown texture is assigned to texture k if $D_k < D_l$ for all $k \neq l$.

5.2 Wavelet Texture Segmentation

Textural features derived from a wavelet decomposition of a test image $I(x, y)$ that are utilised to achieve segmentation are presented by Unser in [134]. Here Unser uses the over-complete wavelet decomposition, or discrete wavelet frames (DWF), where the output of the wavelet filter banks are not sub-sampled. This makes the feature extraction process of the algorithm invariant to translations of the input signal. Texture characterisation is achieved by calculating the average power, $E\{w^2\}$, for each wavelet subband, which equates to the variance if the subband has zero mean. The filterbank is implemented using a pyramid decomposition, see Figure 2.11(a). Therefore, a wavelet decomposition to a depth L results in an N dimension feature vector

$$\begin{aligned} \mathbf{w}(x, y) &= \{w_l(x, y)\}_{l=1\ldots N} \\ &= [s_L(x, y), d_L(x, y), \cdots, d_1(x, y)], \end{aligned} \tag{5.4}$$

where $N = 1 + 3L$, $s_L(x, y)$ is the coarse scale approximation of the input image and $d_l(x, y)$ is a detail image determined by l, see Table 2.1. Note that as $E\{w_1\} = E\{x\}$ and $E\{w_l\} = 0$ for $l = 2, \ldots, N$, it is necessary to adjust the feature values in the low pass channel by subtracting $E\{I\}^2$ which results in an more accurate estimate of the channel variance [134].

The feature extraction technique, based on DWF, is incorporated into a texture segmentation algorithm by [134]. Segmentation is achieved by calculating local estimates of the wavelet coefficient variances, which are used as texture features. The size of the neighbourhood used to calculate the local energy, an estimate of the variance, is dependent on the scale at which the features are extracted. Texture features at a scale m, are computed by convolving the squared (or full wave rectified) output of the wavelet filterbank with a window of dimensions related to the scale m given by

$$f_{l,m}(x, y) = b_m^3 * |w_l(x, y)|^2, \tag{5.5}$$

where b_m^3 is a cubic B-spline kernel enlarged by the integer factor m, which results a window with a close approximation to a Gaussian [134]. There is a need to subtract the mean value from the low frequency subband ($l = 1$) to obtain a more accurate estimate of the true variance.

The wavelet based tree-structured classification method, described in Section 5.1.1, has been extended to achieve texture segmentation [23]. The tree-structured transform is used to calculate dominant frequency channels containing significant textural information, which identifies nodes in the tree decomposition. At every node the local energy is calculated for every pixel using

$$e(x_{i,j}) = \frac{1}{L^2} \sum_{k \in W_{i,j}} |x_k|, \tag{5.6}$$

where $W_{i,j}$ is a rectangular window of dimensions $L \times L$ centred on the pixel position $x_{i,j}$. A suitable value defining the size of the window $W_{i,j}$ is $L = 3$. The energy values are used as local texture features and provide a method for segmentation using the fuzzy c-means algorithm.

5.2.1 Wavelet Based SAR Texture Segmentation

A supervised SAR image segmentation algorithm based in the redundant wavelet transform (see Section 2.4.5) is described in [27]. After the input image is decomposed, each pixel is represented by a 4 element vector whose components are taken from the wavelet subbands. The texture class centres were estimated by taking the mean value in each subband for a given input texture to calculate each texture *codevector* (1.9). Each pixel feature vector was assigned to a texture class by a minimum distance classifier. The Mahalanobis distance function (see Table 1.2) was used to calculate the distance between every pixel feature vector and codevector in the database of reference textures. Results from decomposition level 2 were presented, where the 4 element feature vector was formed by decomposing the LL image from level 1, which resulted in improved segmentation results. The overall scheme of the proposed algorithm is shown in Figure 5.2.

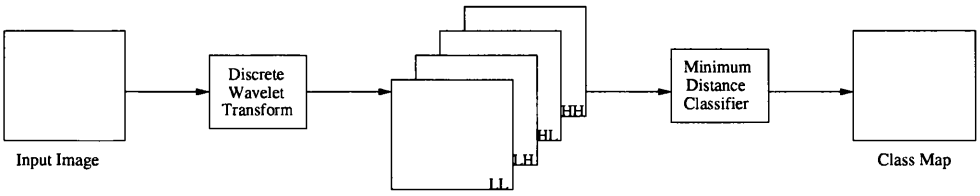


Figure 5.2: Supervised minimum distance wavelet based segmentation scheme.

5.3 Wavelet Packet Segmentation Algorithm

The previous sections of this chapter have shown how image texture features can be derived utilising a wavelet packet decomposition. After processing an input image with a DWT filterbank, a full wavelet decomposition is achieved yielding information on the spectral content in different frequency bands, which provide a mechanism for forming textural feature vectors [39]. As textural information resides within within mid-frequency bands [24], segmentation is achieved by classifying the local texture feature vectors on a per-pixel basis.

It is desirable to use a symmetric wavelet in the texture feature extraction stage of the algorithm as an asymmetric wavelet will introduce a spatial shift in the feature space. Daubechies 9/7 symmetric wavelet has shown to provide good texture feature extraction whilst preserving phase information. The rows and columns of the image that are processed by the DWT filterbank, see Section 2.4.2, can be regarded as input vectors of finite length. When processing vectors of finite length with a multi channel filterbank it is important to handle the boundary conditions in a way to to prevent their effects propagating to the highpass filter wavelet coefficients [21]. Symmetric extensions, see Appendix B, have been implemented in the wavelet texture feature extraction algorithm to prevent distortion of the wavelet coefficients. As Daubechies 9/7 wavelet was used to extract the texture features for the test image, which has odd length filter taps, whole-sample symmetric extensions, see Appendix B, were used to handle the boundaries of the image.

Due to the properties of the wavelet filterbank, see Appendix A, the subbands will have a weighting due to the *regularity condition.* These weightings are illustrated in Figure 5.3, and show how the wavelet coefficients in the subbands need to be adjusted

to allow for this weighting.

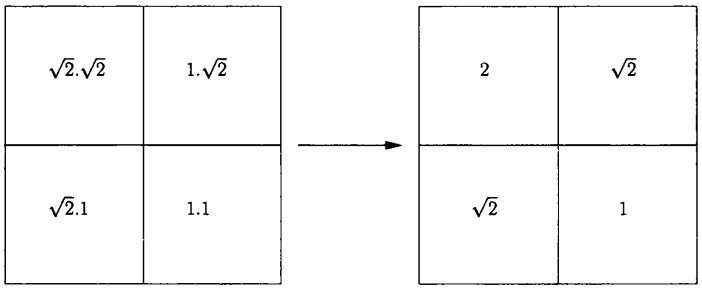


Figure 5.3: Weighting in wavelet subbands.

For the developed wavelet based texture segmentation algorithm to be consistent with the model of the HVS [78] and the Gabor segmentation algorithm described in Section 4.3, the coefficients output by the wavelet filterbank need to be processed with a non-linear operator and have spatial information included with a postfiltering stage. The magnitude operator was selected as the non-linear process, as this proved to be successful in generating the results in Section 4.3. Other texture segmentation algorithms have used Gaussian weighted windows when applied as a postfilter resulted in preservation of texture boundaries [62]. For wavelet based segmentation a cubic B-spline filter which provides a close approximation of a Gaussian weighted window has been used [134]. Using this information a Gaussian weighted window was selected as the postfilter, with the spatial dimensions,

$$\sigma_{post} = \frac{3.2^{scale-1}.\sqrt{2}}{2}, \tag{5.7}$$

where scale is the decomposition level.

A linear post-processing operator will have the undesirable effect of smoothing sharply defined objects in the feature space, thus degrading the definition of texture region boundaries. An alternative method of introducing localised inhibition is to used the watershed transform to identify regions of homogeneous texture and adjust wavelet coefficients that are members of the same region. This should respect texture boundaries in the feature space and lead to a more accurate segmentation. A threshold value

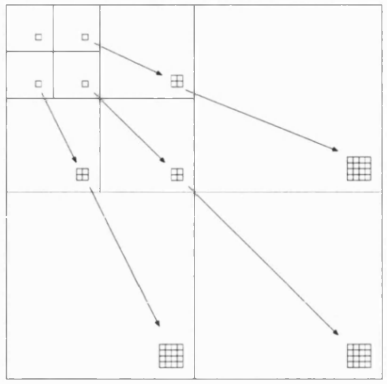


Figure 5.4: Relationship of wavelet coefficient area with scale.

calculated using,

$$t_{ws} = 16^{scale}, \tag{5.8}$$

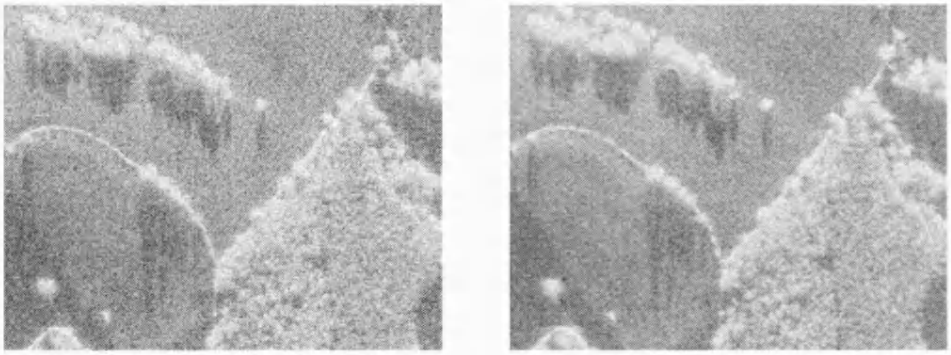
produced regions with areas that were in the same range as described in Section 4.3.

An alternative multi-scale texture segmentation algorithm can be achieved by substituting the redundant wavelet filterbank for a non-redundant wavelet filterbank, which is implemented by sub-sampling the wavelet coefficients after each linear filtering stage (see Figure 2.10).

Prior to the post-processing stage of feature extraction, either a Gaussian postfilter or post watershed feature adjustment, the outputs from the non-redundant DWT filterbanks are resized using nearest neighbour interpolation to the original image size. Figure 5.4 provides a visualisation of the relationship between the area over which a wavelet coefficient conveys information and scale of decomposition. The filtered images are then post-processed with the same parameters for the redundant wavelet transform algorithm (5.7) and (5.8) which are dependent on scale of analysis.

5.3.1 Incorporating Speckle Reduction

Once an image has been represented in the wavelet domain for feature extraction, the features will include components due to scene texture and speckle noise [117]. For a



(a) Test image.

(b) Wavelet shrinkage speckle reduction $\gamma = 2$.

Figure 5.5: Wavelet shrinkage speckle reduction results.

robust feature extraction scheme it is desirable to remove speckle noise components from the feature space, whilst preserving textural information. This will reduce the number of mis-classifications due to the presence of speckle noise.

By studying the properties of SAR images it has been shown in Section 3.3 how textural information and speckle noise are separable in the wavelet domain. When an image has been logarithmically transformed, it is possible to remove the speckle noise component from a subband by a thresholding function in Table 3.2. Once a subband has been thresholded with a function in Table 3.2, it is assumed the speckle noise component has been removed from the subband. Therefore all subsequent daughter subbands in the wavelet packet decomposition are assumed to be noise-free, and do not need to be processed. The subbands to be thresholded to remove the speckle noise are the subbands of the wavelet pyramid transform, Figure 2.11(a), excluding the coarsest scale LL subband [54].

The response of the HH subband at 1 decomposition level can be to contain no textual information and can be wholly attributable to noise. Therefore the noise level σ can estimated by calculating the standard deviation of the wavelet coefficients in the HH subband at level 1. At every increase in decomposition level the threshold value $t = \gamma\sigma$ has to be halved to be consistent with the wavelet coefficients that have been adjusted for the regularity condition.

The presence of speckle only effects wavelet coefficients up to a scale corresponding to speckle grain size [115]. The thresholding of wavelet coefficients above this level would not remove any speckle noise component, whilst having the effect of suppresses textural information. Wavelet thresholding has only been applied to wavelet coefficients up to and including two levels of decomposition. The SAR image used to generate the results, presented in Section 5.4, is shown in Figure 5.5. The test image is shown along with a wavelet shrinkage speckle reduced image, Figure 5.5(b). From analysis of Figure 5.5(b) it is possible to gain an appreciation of the speckle component removed from the feature space.

5.4 Results

Textural information was extracted from wavelet decomposition levels 1—3, as features extracted from decomposition levels higher than level 3 are considered to be of a scale too coarse to contain textural features. Texture analysis using features from the first 3 decomposition levels, results in an 84 element feature vector, see Table 5.1, which is a manageable size to be clustered with the k-means clustering algorithm, described in Section 1.3.3.1.

To test the effectiveness of the multi-scale texture segmentation based on the wavelet transform, algorithms utilising both the redundant and non-redundant forms of the wavelet transform are applied to the Brodatz textured image of Figure 4.3(f). This would provide a direct comparison to the segmentation algorithm utilising a Gabor filterbank described in Section 4.3. Analysis of Figure 5.6 shows the redundant DWT texture segmentation scheme produces better performance at the texture boundaries in comparison to a similar result obtained via a Gabor filterbank, see Figure 4.6. Table 5.2 shows an average of 97.1840% classification accuracy is obtained with redundant wavelet transform features. Results of the segmentation obtained with features extracted via the non-redundant wavelet transform are presented in Figure 5.7, with Table 5.3 showing an average correct classification of 96.2280%, which is comparable to the result obtained via redundant wavelet transform feature extraction.

Results from Figures 5.6 and 5.7 provide verification that the wavelet transform can

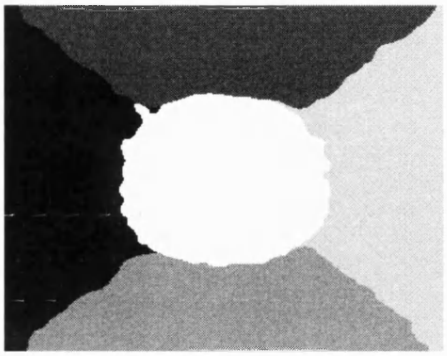


Figure 5.6: Unsupervised segmentation of compound Brodatz texture image utilising features extracted via a redundant DWT filterbank with spatial coordinates.

be utilised for texture feature extraction and implemented as a suitable segmentation algorithm. Therefore wavelet-based techniques were applied to the SAR image in Figure 4.7, using the same number of texture classes for the results generated in Section 4.3. To incorporate speckle noise suppression into textural feature extraction the wavelet coefficients present at the output of the filterbank are modified with the soft thresholding function, see Table 3.2, prior to the non-linear magnitude operator.

A segmentation result utilising redundant DWT texture features with a Gaussian post-filter are presented in Figure 5.8(a). Once a SAR image is represented in the wavelet domain it is possible to suppress the speckle noise components, by the shrinkage of coefficients with a suitable thresholding function, see Section 3.3.1. The resulting speckle reduced segmentation is shown in Figure 5.8(b).

True class:	Classified as:				
	D77	D84	D55	D17	D24
D77	99.60%	0.22%	0.02%	0.00%	0.16%
D84	0.84%	96.71%	0.00%	1.86%	0.58%
D55	1.76%	0.00%	97.56%	0.42%	0.26%
D17	0.00%	0.26%	0.65%	98.80%	0.28%
D24	1.55%	2.02%	1.35%	1.84%	93.25%

Table 5.2: Confusion matrix for unsupervised segmentation of compound Brodatz texture image utilising features extracted via a redundant DWT filterbank with spatial coordinates.

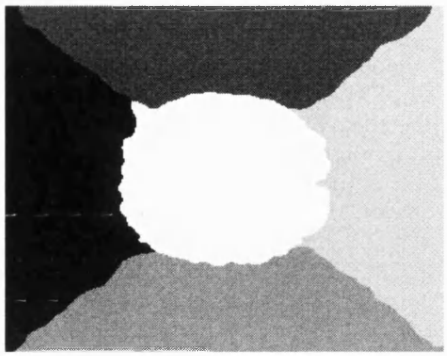


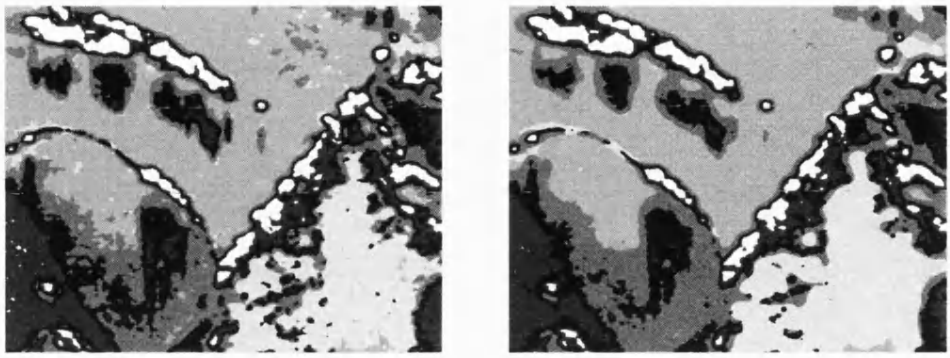
Figure 5.7: Unsupervised segmentation of compound Brodatz texture image utilising features extracted via a non-redundant DWT filterbank with spatial coordinates.

Segmentation results obtained by post-segmentation feature adjustment, as opposed to Gaussian post-filtering, are presented in Figure 5.9. Again speckle suppression is incorporated into the texture feature extraction framework by thresholding the wavelet coefficients prior th the non-linear magnitude operator, the resulting segmentation is shown in Figure 5.9(b).

Results showing the segmentation of Figure 4.7 with a non-redundant wavelet transform are presented in Figure 5.10. As for the redundant wavelet transform, regions identified with the multi-scale watershed segmentation were used to adjust the wavelet coefficients dependent on their local neighbourhood. This has the effect of preserving the boundaries of the texture regions, which can be seen from examination of Figure 5.11.

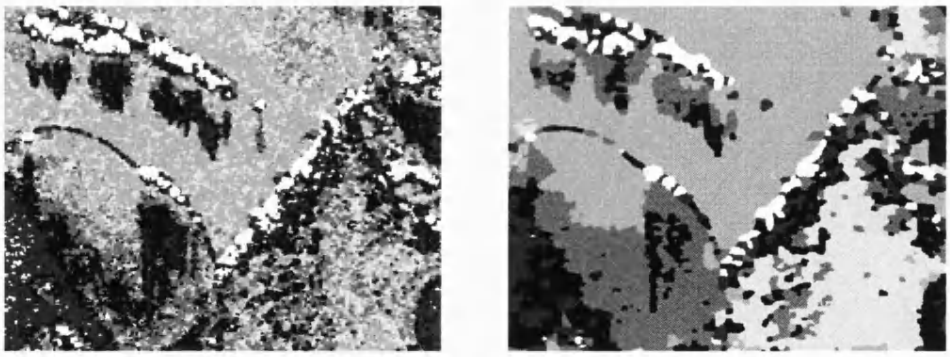
True class:	Classified as:				
	D77	D84	D55	D17	D24
D77	98.72%	0.86%	0.08%	0.00%	0.34%
D84	0.17%	92.88%	0.00%	6.10%	0.85%
D55	1.23%	0.00%	98.38%	0.19%	0.19%
D17	0.00%	0.00%	1.86%	98.07%	0.07%
D24	0.86%	1.49%	2.04%	2.52%	93.09%

Table 5.3: Confusion matrix for unsupervised segmentation of compound Brodatz texture image utilising features extracted via a non-redundant DWT filterbank with spatial coordinates.



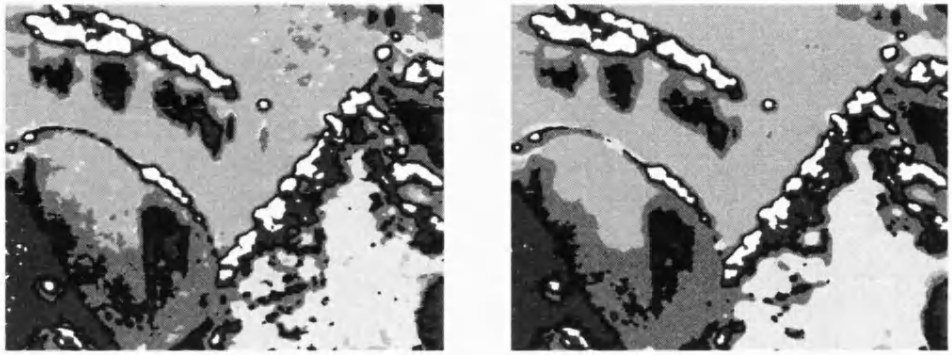
(a) Without speckle reduction. (b) With speckle reduction.

Figure 5.8: Redundant DWT filterbank with Gaussian postfilter unsupervised segmentation results.



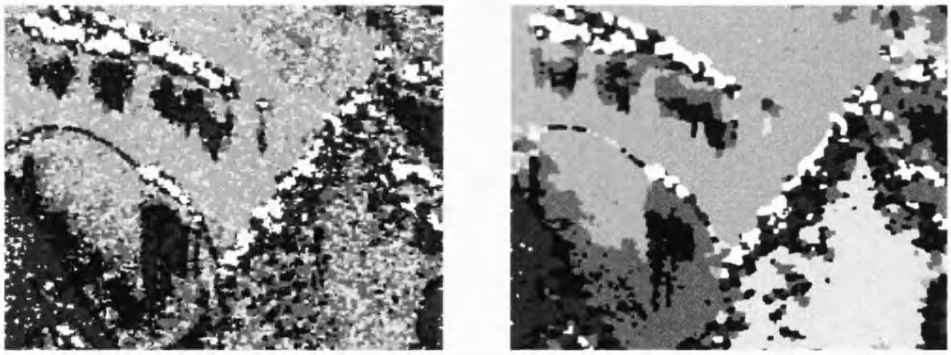
(a) Without speckle reduction. (b) With speckle reduction.

Figure 5.9: Redundant DWT filterbank with post watershed feature adjustment unsupervised segmentation results.



(a) Without speckle reduction. (b) With speckle reduction.

Figure 5.10: Non-redundant DWT filterbank with Gaussian postfilter unsupervised segmentation results.



(a) Without speckle reduction. (b) With speckle reduction.

Figure 5.11: Non-redundant DWT filterbank with post watershed feature adjustment unsupervised segmentation results.

5.5 Discussion

The wavelet based multi-scale texture segmentation algorithm [134] is analogous to the Gabor texture segmentation scheme [62]. After feature extraction is achieved by processing an image with a multi-scale filter bank, a non-linearity operator is applied to the filtered images followed by a postfilter. However the technique presented in [134] utilises the wavelet pyramid decomposition, thus extracting information from the low frequency regions of textured images. Previous wavelet based techniques for the segmentation of SAR imagery [27] also use a pyramid based decomposition of the input image. It has been acknowledged that salient textural information is located in some mid-frequency range of an image [24], thus the pyramid based decomposition is not optimal for texture feature extraction. This chapter has seen the development of a segmentation algorithm utilising the wavelet packet transform, described in Section 2.4.4, for both the redundant and non-redundant wavelet transforms.

A comparison between the performances of the redundant and non-redundant wavelet transforms can be obtained by analysis of the confusion matrices in Tables5.2 and 5.3. This analysis shows the non-redundant DWT produces a segmentation that is nearly as accurate as a segmentaion obtained via the redundant DWT, with a much reduced computational cost. A segmentation algorithm based on the non-redundant DWT is attractive as it fits in with image compression schemes.

The application of wavelet based multi-scale texture segmentation to SAR imagery showed improved results in comparison to results obtained with a Gabor filterbank described in the previous chapter. An advantage of wavelet based feature extraction over Gabor filterbank method is the ability to incorporate into the feature extraction stage. When speckle noise suppression was integrated into the feature extraction framework, for the segmentation of the SAR imagery, this resulted in less intra region mis-classifications for both the redundant and non-redundant wavelet transform algorithm.

Chapter 6

Morphological Texture Analysis

THE previous chapters have described how the extraction of textural features are achieved by decomposing a signal into elements of different scales with a linear filterbank. Textural analysis is then accomplished by extracting the structural information in each subband. An alternative method of decomposition [11], can be implemented by the datasieve, see Section 2.5.1. This alternative decomposition gives an improved representation of scale and position of sharp edged objects. The datasieve in [11] is implemented with a pyramid of median filters. At each stage the result of subtracting the output of the datasieve from the input gives a differential pyramid. This difference is expressed as a set of granules, which are used as a texture descriptor. The scheme is compared with a decomposition based on Gaussian filters, providing a comparison between linear and nonlinear decompositions. The linear filterbank has the characteristic of spreading the frequency components of sharp objects. This implies that, for a linear system, features that are accurately localised in space are poorly localised in frequency. In contrast to a linear filterbank, a morphological decomposition does not spread sharp edged features in the scale space [11].

This chapter starts by describing the morphological image analysis domain known as the *pattern spectrum.* Section 6.1 describes how the idea of the pattern spectrum was conceived as an alternative to linear filterbanks [80] and provides an analogy with frequency spectrum analysis, thus verifying its suitability to texture analysis. The calculation of the pattern spectrum for binary images is detailed in Section 6.1.1, with

the concept being extended to grey-scale images in Section 6.1.2. The application of morphologically defined scale spaces to the field of image classification is investigated in Section 6.2. More specifically, section 6.2.1 researches how mathematical morphology lends itself to the field of texture classification. This section also highlights how texture analysis can be used to classify an image to an instance of time depending on morphologically define texture features. Section 6.3 describes how texture features can be calculated on a local basis and utilised for the segmentation of textured images. Firstly, the segmentation of binary images is addressed in Section 6.3.1, which is then extended to the problem of segmenting grey-scale images in Section 6.3.2. Using concepts from previous research the novel texture features *area morphology local granulometries* are described in Section 6.4. As area morphology local granulometries provide a means of separating intrinsic scene texture from speckle noise components, Section 6.4.1 describes how noise can be suppressed when an image is represented in the feature domain. Segmentation results, presented in Section 6.5, show the progression of algorithms from supervised binary schemes, to grey-scale unsupervised schemes. The developed segmentation algorithm is then applied to SAR imagery, including results incorporating speckle reduction into the textural feature extraction framework.

6.1 Pattern Spectrum

The field of shape-scale image analysis is addressed in [80], which develops a descriptor called the the *pattern spectrum*. The pattern spectrum detects critical scales of object features within an image, and is achieved via the repeated use of multi-scale non-linear filters, and is referred to as a shape-scale descriptor of an image. In a shape-scale descriptor, the term shape refers to an image object conveying pictorial information. Scale is defined as the size of the shape pattern used to analyse the image.

Traditionally multi-scale image analysis has been achieved by the linear convolution of an image with Gaussian weighted windows. Each Gaussian window is parameterised by the standard deviation σ, which defines the scale at which the image is analysed. The linear filtering approach to multi-scale image analysis gained popularity because of its:

1. Mathematical flexibility;
2. Relationship with Fourier analysis;
3. Plausibility for modelling HVS, see Section 2.1.

However, linear image analysis suffers from the following weaknesses [80]:

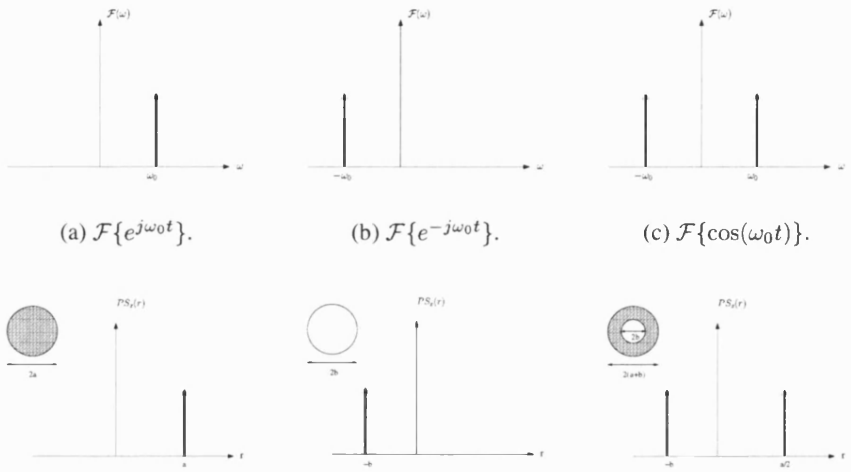
1. Linear filters shift and blur image features;
2. Scale parameter (width of impulse response) is not related to size-based definition of scale;
3. The resultant filtered images do not correspond to the shape of image features.

An alternative to linear filters is the decomposition of an image with a class of non-linear filters. Suitable filters are morphological operators, see Section 2.5. Morphological operators are attractive because they are directly related to image shape, and can be used to define a shape-size descriptor, the *pattern spectrum.*

The analogy between spectral and morphological analysis describes how the pattern spectrum can be used as a signal texture descriptor. The spectral component with a frequency ω, in a signal $x(t)$, can be calculated by multiplying $x(t)$ with the complex sinusoid $\exp\{-j\omega t\}$. The spectral content can then be measured by calculating the area under the modulated signal $x(t)\exp\{-j\omega t\}$ [80]. The frequency spectrum of the signal is obtained by varying the frequency ω, which describes the Fourier transform (2.5). The Fourier transform can be described by an abstract view, where the sinusoid $\exp\{-j\omega t\}$ is a probing pattern that interacts with the signal $x(t)$. Information can then be extracted by a transform, via modulation, and measuring the transformed signal. Although the Fourier spectrum has many uses in the field of signal and image processing, its usefulness with respect to shape-scale image analysis is limited. However, concepts derived from the aforementioned abstract view can be used to develop a *pattern spectrum* for the analysis of images and signals. Let the signal $x(t)$ be replaced by the 2D set $I(x, y)$ representing an image; the complex sinusoid $\exp\{-j\omega t\}$ be replaced by the compact 2D set B_k which depends on the scale parameter k; and the signal modulation be replaced by a morphology based shape-size transform of

$I(x, y)$. The area of the transformed image can be used to produce a pattern spectrum of $I(x, y)$, which measures the size distribution in the image, relative to the shape of the structuring element B_n.

Analogies between the Fourier and pattern spectrum are represented pictorially in Figure 6.1. The pattern spectra in Figures 6.1(d)–(f) are generated with a disk structuring element with increasing radius r. For an input images of a disk, a circle, and a disk with a hole, these are comparable to the Fourier transforms of $e^{j\omega_0 t}$, $e^{-j\omega_0 t}$ and $\cos(\omega_0 t)$ respectively. For a detailed description of how pattern spectra can be calculated, the reader is refereed to Section 6.1.2.



(a) $\mathcal{F}\{e^{j\omega_0 t}\}$. (b) $\mathcal{F}\{e^{-j\omega_0 t}\}$. (c) $\mathcal{F}\{\cos(\omega_0 t)\}$.

(d) Disk. (e) Circle. (f) Disk with hole.

Figure 6.1: Analogies between Fourier and pattern spectra [80].

6.1.1 Binary Image Pattern Spectrum

The opening of a binary image image I with a binary structuring element B (D.7) is defined as the union of all translations of B which are subsets of I, see Appendix D. As the opening operator is an increasing operator when B_1 is a subimage of B_2, then $X \circ B_1$ is a subimage of $X \circ B_2$. Therefore, if $B_1, B_2, B_3, \ldots B_k$ is an increasing sequence

of structuring elements, the filtered images will form the decreasing sequence:

$$X \circ B_1 \subset X \circ B_2 \subset X \circ B_3 \subset \ldots \subset X \circ B_k \tag{6.1}$$

The total number of pixels remaining in each successive opening results in the decreasing function, $\Psi(k)$, where for a given value of K, $\Psi(k) = 0$ for $k \geq K$. Various textural information can be extracted from $\Psi(k)$ depending on the shape of the structuring element. The sequence of images

$$I_k = \{X \circ B_k : 1 \leq k \leq K\}, \tag{6.2}$$

defines the granulometry, and the resulting function $\Psi(k)$ is known as the size distribution [35].

Since $\Psi(k)$ is a decreasing function, the normalisation (6.3) yields the probability distribution function,

$$\Phi(k) = 1 - \frac{\Psi(k)}{\Psi(1)}. \tag{6.3}$$

The discrete derivative (6.4) of the probability distribution function $\Phi(k)$ produces the discrete density function

$$\delta\Phi(k) = \{\Phi(k) - \Phi(k-1) : 1 \leq k \leq K\}. \tag{6.4}$$

A local granulometric size distribution can be generated by placing a window W_x, at each image pixel position x. After each opening operation, a count of the number of remaining pixels within W_x is calculated. This results in a local size distribution, Ψ_x [34]. Each local size distribution can be normalised in the same manner as the global granulometry giving the local distribution

$$\Phi_x(k) = 1 - \frac{\Psi_x(k)}{\Psi_x(1)}. \tag{6.5}$$

The density $\delta\Phi_x$ yields the local pattern spectrum at pixel x, and provides a description of local texture. Textural features that can be extracted from $\delta\Phi_x$ are the moments of

the function.

6.1.2 Grey-Scale Image Pattern Spectrum

The extraction of the pattern spectrum from grey-scale images is addressed in [80]. The pattern spectrum for grey-scale images can be described by the decreasing sequence

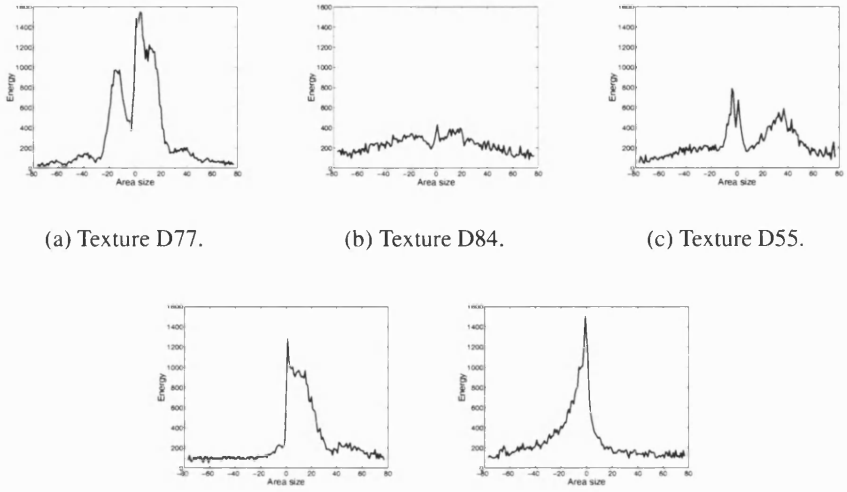
$$PS_f(B_{+k}) = A[f \circ B_k - f \circ B_{k+1}], \tag{6.6}$$

where $A[f] = \sum_{xy} f(x, y)$. The size parameter is defined by k, while the structuring element B defines the shape parameter. By keeping B fixed (6.6) produces a *size histogram* of f relative to B.

This alternative method of calculating the pattern spectrum yields a result similar to the pattern spectrum for binary images (6.3). The sequence of morphological openings produces a decreasing sequence reminiscent to (6.1). The discrete differentiation is replaced with the pointwise difference of functions, producing residual grain images. The pattern spectrum can be extended to contain negative values by replacing the opening operator in (6.6) with a morphological closing, given by

$$PS_f(B_{-k}) = A[f \bullet B_k - f \bullet B_{k-1}]. \tag{6.7}$$

By varying both the structuring element B, and the size parameter, k, the full shape-size spectrum of $f(x, y)$ can be achieved. A disadvantage of generating pattern spectra with structuring element-based morphology is the vast amount of structuring elements required to capture the shapes of all image features. A full shape-scale spectrum can be achieved by utilising area morphology, see Section 2.5.2, as for a given size parameter all shapes contained within the size can be obtained, see Table 2.2. Examples of area pattern spectra calculated using (6.6) and (6.7), for the textured images in Figures 4.3(a)–(e), are shown in Figure 6.2. Analysis of the pattern spectra trends show how they could be implemented to provide texture discrimination.



(a) Texture D77. (b) Texture D84. (c) Texture D55.

(d) Texture D17. (e) Texture D24.

Figure 6.2: Grey-scale image area pattern spectra.

6.2 Morphological Image Classification

When classifying pixels within an image not only is the intensity of an object an important measure, so is the scale at which the object occurs. Morphologically derived scale-spaces provide a means for extracting information from an images at a number of scales, which may result in more accurate classification results in comparison to fixed scale classification algorithms. Also, the distribution of image features at multiple scales can also be implemented as descriptors of image texture.

Area morphology has been used to define a scale space which is applied to image classification [2]. A scale space is created by the successive application of an area open-close (AOC) or an area close-open (ACO) morphological operation. The pixel positions in the scale space correspond to the same spatial location in the image under analysis, thus creating a local scale space vector for every pixel. The scale space feature vector contains a series of intensity values for a given set of scales, caused by image granulometry.

Image classification techniques utilising the original input image at a fixed scale can result in classification errors. Typical errors are intra-object classification errors, when pixels belonging to the same object class are assigned more than one object class. Single scale classification techniques can be sensitive to localise image noise, which can introduce changes in intensity, but are of a limited area. Acton and Mukherjee [2] propose an image classification technique, based on morphological image scale space. The scale space is defined as a set of image representations to detect image objects at different scales. A scale space vector is obtained for every pixel in the image that describes the pixel intensity through a range of scales. The area open-close (AOC) and area close-open (ACO) scales spaces are used to obtain the scale space representation of an image. Scale spaces obtained via AOC and ACO operators do not introduce new features at increasing scales, hence, the introduction of new regions are avoided due to processing artefacts. For an input image I, the scale space is defined as the set $\{\mathbf{I}\}$ of scale representations I_s. The AOC scale space is calculated by

$$I_{s(\alpha)} = I_{s(\alpha-1)} \circ B_\alpha \bullet B_\alpha, \tag{6.8}$$

where B_α describes the family of SE's to contain all shapes formed by α connected pixels. Similarly the ACO scale space is obtained by

$$I_{s(\alpha)} = I_{s(\alpha-1)} \bullet B_\alpha \circ B_\alpha. \tag{6.9}$$

Area morphology-based scale spaces are attractive in the field of image classification as the input image is simplified and intra-region homogeneity is increased with increased scale [2]. For a given scale space, $\{\mathbf{I}\}$, the intensity at pixel x, y, at scale s, is defined as $I_s(x, y)$. For a fixed pixel position (x, y), $\mathbf{I}(x, y)$ denotes the one-dimensional scale space vector that represents the scale space evolution of $I(x, y)$. The scale space vector can be used to classify image objects as pixels with similar scale space vectors will form clusters in the scale space. For the scale space classifier [2] it is assumed that pixels should be classified by intensity at a range of scales, and that each pixel is a member of a particular object at every scale. Therefore $I_s(x, y)$ is the intensity of an object at scale s at pixel position (x, y).

Area morphology scale space classifiers have been shown to give more accurate results in comparison to fixed scale classifiers. This can proved by the example when two pixel

positions are members of the same object class, but have different initial intensities, *i.e.* $I_0(x_1, y_1) \neq I_0(x_2, y_2)$, an area morphology based classifier with cluster the two pixels more closely than a fixed scale classifier [2].

6.2.1 Morphological Texture Classification

Morphological granulometric size distributions provide a way of describing image granularity or *texture*. Chen *et al.* investigated the classification of local granulometric size distributions for stationary textured images [25]. The pattern spectrum described in Section 6.1 is calculated over a window for grey-scale images of homogeneous texture. A textural feature vector results from the moments of the local granulometric size distributions. If an image is regarded as a random process, the local size distributions are random functions, with their moments as random variables that can be used for classification.

A family of structuring elements is generated by taking a primary function B and iteratively applying a morphological dilation,

$$B_{k+1} = B_k \oplus B. \tag{6.10}$$

Using a set of increasing size structuring elements, for an image I, generates a set of morphologically opened images $\{I \circ B_k\}$ that is a decreasing sequence. Once a series of morphologically filtered images has been obtained, local granulometries can be generated by calculating the volumes within a window about each pixel, which results in the decreasing function $\Psi(k)$. Normalisation of $\Psi(k)$ (6.3) produces a probability distribution function. As for the binary pattern spectrum, discrete differentiation of $\Phi(k)$, using (6.4), yields a probability density function $\delta\Phi(k)$, whose moments can be used as texture descriptors.

The research presented in [25] also provides an investigation into the use of both opening and closing granulometries as texture descriptors. Closing granulometries can be defined as the dual of opening generated granulometries. Using the closing operator (2.18), structuring element sequences are constructed such that, $I \bullet B_{k+1} \geq I \bullet B_k$. Opening granulometries contain information obtained from probing beneath the im-

age surface, whilst closing granulometries contain information obtained from probing above the image surface. Two classifiers are implemented in [25], one containing texture features from opening granulometries, whilst the other utilised texture features obtained via opening-closing granulometries. This enables the investigation into the improvement in classification performance if a combination of opening and closing morphological operations are implemented.

Texture classification is achieved by comparing the set of granulometric moments that represent an unknown texture with a *codebook* of reference textures. The input images are decomposed using seven different structuring elements: cone-shaped, ball-shaped, flat-top, horizontal linear, vertical linear, positive diagonal linear and negative diagonal linear. For each structuring element both an opening and closing granulometry was calculated, with the first three central moments being uses as texture features. This results in a 42 element feature vector for each texture. Each classifier was trained with the feature vectors of 300 randomly positioned pixels for each textured image. A Gaussian maximum likelihood classifier was implemented by calculating the mean feature vector (1.9) and covariance matrix (1.10) for the training data in each texture class. Once the training data had been obtained, the classification algorithm was tested on 500 random pixel positions for each texture.

Analysis of results obtained with opening granulometries and opening-closing granulometries showed improved texture discrimination can be achieved by including information from both above, and below the image surface in the feature space. It has been shown for a reduced feature set, relatively high classification results can be achieved [25].

Textured images have been classified with the use of texture features derived from the morphological size density of an image [119]. A granulometry is obtained is obtained by the recursive application of morphological openings to an image with increasing size structuring elements. The morphological dual of a granulometry, known as an *anti-granulometry*, is obtained by the application of closings with increasing size structuring elements to an image [120]. A granulometry and associated anti-granulometry are used to provide a multi-resolutional decomposition of an image $I(x, y)$. However, the decomposition is not in terms of frequency, which is a custom of linear multi-resolution techniques, but in terms of size of components in $I(x, y)$. A multi-resolution

decomposition of an image can be obtained by the *size distribution* (6.11). For an input image I with grey-levels $\{0, 1, 2, \ldots, G-1\}$ the size distribution is calculated by,

$$S_I(k) = \frac{1}{G|W|} \begin{cases} E[\|I \circ B_k\|_W], & \text{for } k \geq 0 \\ E[\|I \bullet B_k\|_W], & \text{for } k \leq -1, \end{cases} \tag{6.11}$$

where $E[X]$ is the expected value of X, $|X|$ denotes cardinality of a set X, $\|X\|_W = \sum_{w \in W} |X(w)|$ and W is a bounded observation window.

For texture classification it is desirable to represent an image by means of a decomposition of detail images $(I \circ B_k) - (I \circ B_{k+1})$ and $(I \bullet B_k) - (I \bullet B_{k-1})$ [120]. This decomposition of the detail images gives rise to the *size density* of an image (6.12).

$$s_I(k) = \frac{1}{G|W|} \begin{cases} E[\|I \circ B_k - I \circ B_{k+1}\|_W], & \text{for } k \geq 0 \\ E[\|I \bullet B_k - I \bullet B_{k-1}\|_W], & \text{for } k \leq -1. \end{cases} \tag{6.12}$$

Texture classification based on features derived by morphological size distributions, produces highly accurate results, even when texture classes are visually similar [119].

Traditional texture classifiers compare a calculated feature vector against a predefined *codebook* of known textures, and assign the closest class to the texture under analysis. An alternative implementation for texture classification is to classify an observed image to an instance of time.

Using mathematical morphological, texture analysis is applied to an image of a corroded component, to quantify weakening of the component [84]. A grey-scale granulometry is calculated by applying a series of morphological openings (D.7) to an input image with an increasingly large *SE*. The energy removed after each opening with a given SE is normalised to calculate the pattern spectrum [34]. Moments of the pattern spectrum are then calculated and used as texture features in the classifier. Using simulated images, which the corrosion parameters are known, a relationship between the pattern spectrum moments and parameters defining the level of corrosion can be established. Once the classifier has been trained using simulated images, granulometric moments derived from input images can be used to estimate the level of corrosion of the components in the images.

6.3 Morphological Texture Segmentation

Local granulometric size distributions are generated by performing a granulometry on an image and measuring pixel values, within the local neighbourhood of each pixel, at each successive morphological scales. Normalisation of the calculated size distribution (6.5) provides a probability density for every pixel, which contains textural information local to each pixel position. The calculation of a probability distribution for every pixel position provides a means of classification according to the moments of the densities. Local granulometric size distributions have been used as texture features in a pixel classification scheme [35].

6.3.1 Binary Image Segmentation

Segmentation of binary images using local granulometric size densities has been applied in [32, 33, 34, 35]. Morphological granulometries are generated by the successive opening of an image with an increasing sequence of structuring elements. The resulting set of images is a decreasing sequence where each image is a subimage of the previous. A local size distribution is calculated by taking a pixel count within a local window at each stage of the granulometry. Normalisation of the local size distribution, using (6.3), gives the distribution function, whose discrete derivative $\delta\Phi(k)$ (6.5) produces the *pattern spectrum.*

For a binary input image the granulometry in [32] was generated based on a structuring elements sequence of disks with increasing radii. The procedure of [32] is to find the mean value of the pattern spectrum at each pixel position. This produces a grey-scale image known at the *pattern spectrum mean* (PSM). Segmentation is achieved by thresholding the PSM image with an suitable value.

A supervised classification scheme can be implemented by comparing the observed moments at each pixel position, in a binary image, to a database of texture moments [35]. Texture feature extraction can be provided by employing several structuring element sequences to generate a number of granulometries. Each granulometry then provides different texture qualities resulting in a more robust segmentation. The moments of

$\delta\Phi(k)$ (6.5) can be used to extract textural information.

Local granulometric size distributions have been used to segment simulated SAR images based on texture [33]. In this research an image was analysed with 4 linear structuring elements (vertical, horizontal, positive diagonal and negative diagonal), along with a circular structuring element. The 5 structuring elements used to create the local size distributions are B_V, B_H, B_P, B_N, B_C. For each local size distribution generated the first order moment, μ_x, was calculated (6.15).

The 5 element vector, μ_x for each structuring element, yielded different values for the regions of texture in the image under analysis. Segmentation was then attempted by combining the the textural information from the 5 structuring elements to ease the differentiation of local textures. Regions of maximum linear dimension, regardless of direction were calculated by the $MaxLin$ feature

$$MaxLin = M_x = \max(\mu_{Vx}, \mu_{Hx}, \mu_{Px}, \mu_{Nx},). \tag{6.13}$$

The $MaxLin$ feature alone still does not differentiate between regions of large linear components and circular components with similar diameters. To differentiate between these regions the $Linearity$ feature

$$Linearity = L_x = \frac{M_x}{\mu_{Cx}}, \tag{6.14}$$

was used (6.14). The $Linearity$ ratio is unity for circular features, regardless of diameter. The ratio increases for textural features whose ratio of linear length to width increases [33].

For the supervised texture segmentation algorithm [34] the same 5 structuring elements, B_V, B_H, B_P, B_N, B_C, were used to create the granulometries. For each granulometry the first 3 central moments of the local pattern spectrum are used as textural features. As the local pattern spectrum $\delta\Phi_x(k)$ is a discrete probability density function the first 3 moments of the set can calculated [55]. The features are the pattern spectrum mean,

$$PSM_x = \mu_x = \sum_{k=1}^{K} \delta\Phi_x(k)k, \tag{6.15}$$

the pattern spectrum standard deviation,

$$PSSD_x = \sigma_x = \sqrt{\sum_{k=1}^{K} \delta\Phi_x(k)(k-\mu_x)^2}, \tag{6.16}$$

and the pattern spectrum skewness,

$$PSS_x = \sum_{k=1}^{K} \delta\Phi_x(k) \frac{(k-\mu_x)^3}{\sigma_x}. \tag{6.17}$$

Skewness is a measure of symmetry, or more precisely, the lack of symmetry of a data set. The skewness for a Gaussian distribution is zero, and any symmetric data should have a skewness near zero. A negative value of skewness indicates data that is skewed left and positive value of skewness indicates data that are skewed right.

In addition to the 3 features derived from the pattern spectrum for each structuring element, the $MaxLin$ and $Linearity$ features were included. This resulted in a 17 element feature vector for each pixel position. The database of codevectors used for the supervised classifier were generated by calculating the 17 feature images for a region of homogeneous texture, then calculating the mean of each feature image, $\mu_{PSM}, \mu_{PSSD}, \mu_{PSS}$, for each structuring element sequence and μ_{ML}, μ_{LIN}. Segmentation is then achieved with a minimum distance classifier. The Bayesian distance, see Table 1.2, is calculated between the feature vector at every pixel position and every entry in the database of codevectors. The texture class with the minimum distance is then assigned to the pixel position.

6.3.2 Grey-Scale Image Segmentation

The reason for applying the texture analysis to binary images in [33, 34, 35] is computational time. The analysis of grey-scale images is preferable as more information is available in the classification algorithm.

Segmentation using granulometric texture features has been applied to mammogram

images [10]. A morphological granulometry is generated by applying a sequence of morphological openings with structuring elements of increasing size to the original image. A morphological opening has the effect of reducing the volume beneath the opened image. The rate at which the volume is removed is dependent in the morphological image texture with respect to the structuring element probing the image. Conversely, an anti-granulometry can be generated by applying a sequence of morphological closings to an image. Morphological closings increase the volume below the closed image. Hence, the rate at which volume is increased serves as a texture descriptor. The variation of the image volume against a set of increasing size structuring elements is known as a *size distribution.* Normalisation of this size distribution (6.3) results in a probability distribution function $\Phi(k)$. Discrete differentiation of the probability distribution function (6.4) results in a probability density function or *pattern spectrum.* For the purpose of image segmentation, image texture must be regarded as a local property, therefore any changes to image volume must be computed locally within a window W.

For the texture features used in [10], morphological granulometries are generated using ten different structuring elements. For each structuring element openings and closings are calculated, resulting in two pattern spectra per structuring element. For each pattern spectra the first three moments, mean (6.15), standard deviation (6.16) and skewness (6.17) are calculated and used as texture features. For ten structuring elements this results in a 60 element feature vector, which can be utilised as a local texture descriptor. The window W defining the local region of texture was chosen to be a circle of diameter 15 pixels. In addition to the 60 granulometric texture features, the mean grey level in the window W is used as an extra feature. Segmentation is then achieved by clustering the texture feature vectors with the K-means clustering algorithm, see Section 1.3.3.1.

Greyscale SAR images have been segmented into regions of homogeneous texture with the use of morphological features [74],using morphological openings and closings, to analyse the texture in an airborne SAR image for crop classification. The texture feature set is described by granulometric distributions with the SAR image, which are characterised by residues of morphological filtering operations. Directional structuring elements, as opposed to square ones, resulted in higher classification accuracy. This increase is because directional structuring elements extract information related to

image texture. Linear structuring elements, in both horizontal and vertical directions, were used for texture feature extraction [74].

Once each pixel in the image is characterised by a set of texture features, the classification of the pixels will produce a texture classmap. Prior to classification, each feature image is normalised by,

$$\hat{I}_f(i,j) = \frac{I_f(i,j)}{\max I_f}, \tag{6.18}$$

where $I_f(i,j)$ and $\hat{I}_f(i,j)$ are the grey levels of a pixel in the feature image I_f before and after normalisation respectively. The normalisation process ensures the same brightness degree for all feature images [74].

The classifier implemented is a supervised classification scheme. In the training stage, 20 % of the total pixels for each texture class are used to generate a texture codevector. In the test stage of the classifier each pixel feature vector is given a 3x3 pixel bias, achieved by averaging pixel feature vectors within a local 3x3 neighbourhood. This pixel bias has the result of reducing mis-classifications due to the presence of speckle noise. A post-processing operation, a majority filter calculated over a 3x3 pixel window, was applied to the classmap to reduce mis-classifications. The post-processing operation was applied iteratively to the classmap until idempotence.

6.4 Area Morphology Local Granulometries

In terms of texture classification, area morphology scale spaces are attractive as they can capture all the structures contained in an image. This is advantageous as textured images generally contain far more structures that can be described by a family of fixed structuring elements. In addition, area scale spaces have the property of strong causality which ensures that edge positions are preserved through scale [2]. Successful feature classification was demonstrated by Acton and Mukherjee using a scale space vector containing the intensity of each pixel at a selected set of scales [2].

To extend the use of area morphology operators to texture segmentation, local area morphology granulometries have been implemented for texture feature extraction [41].

The area operators, described in Section 2.5.2, are applied to a local window to produce a single pattern spectrum containing information of all shapes at each scale, given by

$$s_I(\lambda) = \frac{1}{A[I]} \begin{cases} A[\gamma^a_\lambda(I) - \gamma^a_{\lambda+1}(I)], & \text{for } \lambda \geq 1 \\ A[\varphi^a_{-\lambda}(I) - \varphi^a_{-\lambda+1}(I)], & \text{for } \lambda \leq -1. \end{cases} \tag{6.19}$$

A property of area morphology local granulometric texture features is that there is no orientation selectivity in the feature space, thus providing a rotation invariant feature extraction. This is not compliant with texture features derived from Gabor and Wavelet linear filterbanks. However, real-world textures are often not uniform and can occur at arbitrary rotations, making the property of rotation invariance attractive [92].

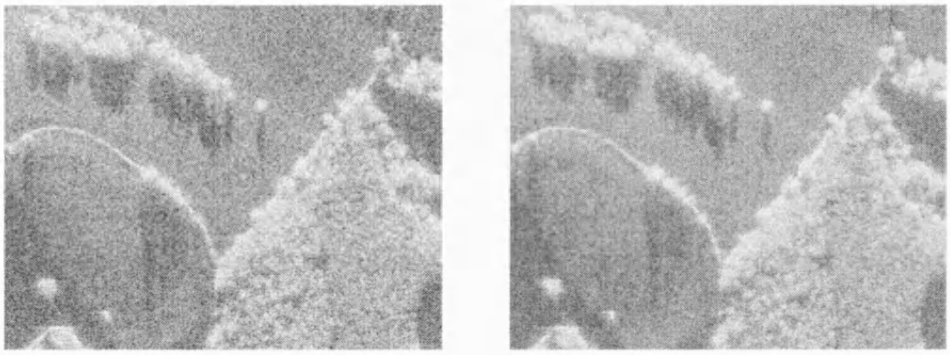
The area pattern spectrum (6.19) has similar properties the model of the human vision system [78], described in Section 2.1.1. Analysis of an image at a given scale results in two responses, one containing information about local maxima, the other containing information about local minima. This is analogous with the half wave rectification process of the vision system model [78].

In practice a maximum area size can be chosen, above which features are considered too large to contain textural information. This approach is computationally simpler than the scale space sampling of [2]. In addition, as the single pattern spectrum produced by this approach contains a limited number of elements it can be used directly to provide a feature vector for classification purposes, instead of indirectly through pattern spectra moments [34, 35].

6.4.1 Incorporating Speckle Reduction

Development of the granulometric shrinkage speckle noise suppression algorithm in Section 3.4.5 highlighted the fact that after an image has been processed by a morphological datasieve, the individual grain images will possess both image features and noise components. Analysis of results in Section 3.5 show that by applying a threshold function to the grain images, a proportion of the speckle noise component can be removed.

From the results presented in Section 5.4, speckle was only removed in the subbands belonging to the first 2 levels of the wavelet decomposition, which correspond to an area size of upto 16 pixels in the original image, see Figure 5.4. Therefore the only the grain images below the size of 16 pixels, $\{G_1, G_2, \ldots, G_{16}\}$ are process with the soft thresholding function. A visualisation of the original and speckle suppressed images are shown in Figures 6.3(a) and 6.3(b) respectively, providing the reader with an interpretation of the different feature space representations.



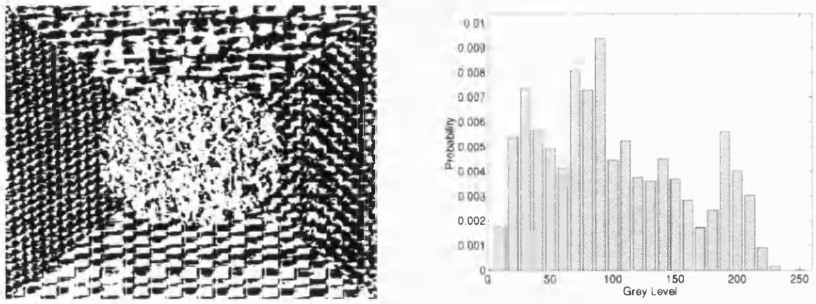
(a) Test image.

(b) Granulometric shrinkage speckle reduction $\gamma = 2$.

Figure 6.3: Granulometric shrinkage speckle reduction results.

6.5 Experimental Results

To provide a set of comparative results, the test image in Figure 4.3(f), was first segmented using binary and grey-scale versions of the local granulometry segmentation scheme described in [34]. The binary test image, Figure 6.4(a), was created by thresholding Figure 4.3(f) so that equal numbers of pixels were assigned to each value. The probability density function, Figure 6.4(b), indicates a suitable threshold value to generate a binary representation of the image. A threshold value of 91 was chosen which resulted in 50.2426% of the pixels in the binary image being set to the value 1.



(a) Binary test image. (b) Histogram.

Figure 6.4: Test image and histogram.

For both the binary and grey-scale methods five structuring elements, four linear (vertical, horizontal, positive diagonal and negative diagonal) and one circular, were used to create local size distributions using a 33×33 window. The maximum size of each structuring element which removed all pixels from the binary image are shown in table 6.1. The first 3 central moments of the local pattern spectra were used as textural features, augmented by two additional features, $MaxLin$ (6.13) and $Linearity$ (6.14), giving a 17 element feature vector.

Structuring element	*Maximum size*
B_V	56 pixels
B_H	41 pixels
B_P	41 pixels
B_N	43 pixels
B_C	5 pixels

Table 6.1: Maximum size of structuring elements.

The database of codevectors used for the supervised classifier were obtained by generating the 17 feature images for homogeneous texture and then calculating the mean feature vector (1.9 and covariance matrix (1.10) for each feature image. The supervised segmentation results achieved by the binary and grey-scale schemes using a minimum distance classifier are shown in Figure 6.5 and Figure 6.6 respectively.

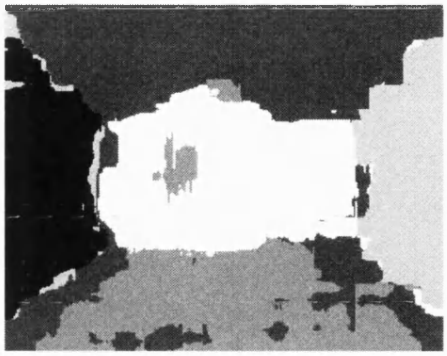


Figure 6.5: Supervised segmentation of compound Brodatz texture image utilising binary pattern spectrum features.

By replacing the binary pattern spectrum in the segmentation algorithm [34] with the greyscale pattern spectrum [80] local size distributions can be calculated for grey-scale images. The pattern spectrum is extended to include negative values with the anti-granulometry using morphological closings. A local granulometric size distribution was generated for each pixel, using the same structuring elements for the binary technique (see Table 6.1) and their moments extracted, along with the $MaxLin$ and $Linearity$ features. Segmentation results for a grey-scale image is presented in Figure 6.6.

The segmentation result for the supervised area morphology local granulometry scheme is given in figure 6.7. Here, the maximum area size was set to 78 pixels, a value equal to the area of the largest structuring element in the previous results, see Table 6.1. The window size which the local granulometry is calculated was 33×33, the same as the

True	Classified as:				
class:	D77	D84	D55	D17	D24
D77	61.91%	24.73%	0.13%	5.34%	7.90%
D84	0.00%	97.44%	1.20%	0.00%	1.36%
D55	0.00%	18.86%	80.74%	0.00%	0.40%
D17	0.00%	26.18%	1.40%	59.30%	13.12%
D24	0.00%	2.67%	10.83%	0.00%	86.50%

Table 6.2: Confusion matrix for the supervised segmentation of compound Brodatz texture image utilising binary pattern spectrum features.

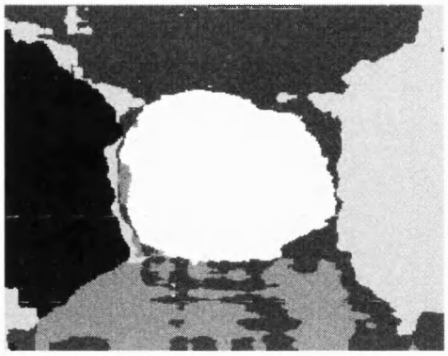


Figure 6.6: Supervised segmentation of compound Brodatz texture image utilising grey-scale pattern spectrum features.

previous results. The granulometry and anti-granulometry were found using (6.19) resulting in a 154 element local size density.

Analysis of Figures 6.6 and 6.7 shows that texture features derived from area pattern spectra improves the segmentation process by an average of 6.8%, in comparison to a scheme based on multiple pattern spectra obtained via a number of structuring elements. This is due to a reduction in inter-region classification errors, and improved results at texture boundaries. To compare the effectiveness of area morphology local granulometries for feature extraction with the Gabor (see Section 4.3) and Wavelet (see Section 5.4) schemes, an unsupervised segmentation scheme using k-means clustering was implemented. Prior to clustering each feature vector is normalised to have a zero mean and constant variance [62], with the inclusion of each pixel's spatial co-ordinates as two extra features. Results of the segmentation along with the confusion matrix are

True	Classified as:				
class:	D77	D84	D55	D17	D24
D77	82.98%	3.94%	1.25%	11.26%	0.56%
D84	0.00%	92.06%	0.04%	6.25%	1.64%
D55	0.24%	37.66%	60.27%	1.32%	0.51%
D17	0.00%	12.12%	0.01%	87.27%	0.61%
D24	0.00%	5.07%	1.59%	0.17%	93.17%

Table 6.3: Confusion matrix for supervised segmentation of compound Brodatz texture image utilising grey-scale pattern spectrum features.

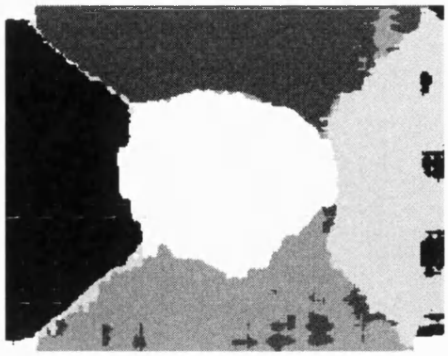


Figure 6.7: Supervised segmentation of compound Brodatz texture image utilising area pattern spectrum features.

shown in Figure 6.8, showing good intra region classification results with an average of 93.2620% of pixels being correctly classified.

The results presented in Table 6.5 show the potential benefits of the new area morphology local granulometry texture segmentation technique. To confirm that the novel feature extraction process of area morphology local granulometries serves as valid texture descriptors, a more rigorous segmentation test was applied using the Oulu database of textures [91].

Each 512×512 pixel test image in the database comprises of 5 compound textures, an example of which is shown in Figure 6.9(a). The reference image indicating the correct location of the texture classes is presented in Figure 6.9(b), and illustrates irregular texture boundaries, that provide a challenging segmentation problem. Table 6.6 shows

True	Classified as:				
class:	D77	D84	D55	D17	D24
D77	96.56%	0.65%	0.00%	0.79%	2.00%
D84	0.32%	92.37%	3.37%	0.99%	2.95%
D55	0.69%	9.03%	79.05%	6.77%	4.46%
D17	6.84%	3.62%	2.18%	86.15%	1.20%
D24	1.10%	0.50%	2.79%	0.00%	95.62%

Table 6.4: Confusion matrix for supervised segmentation of compound Brodatz texture image utilising area pattern spectrum features.

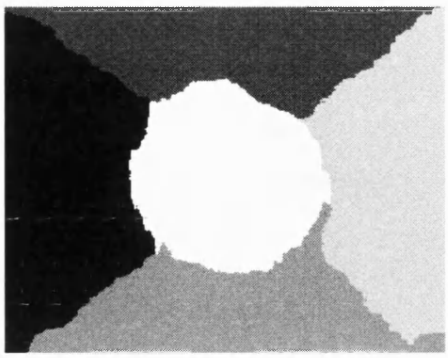


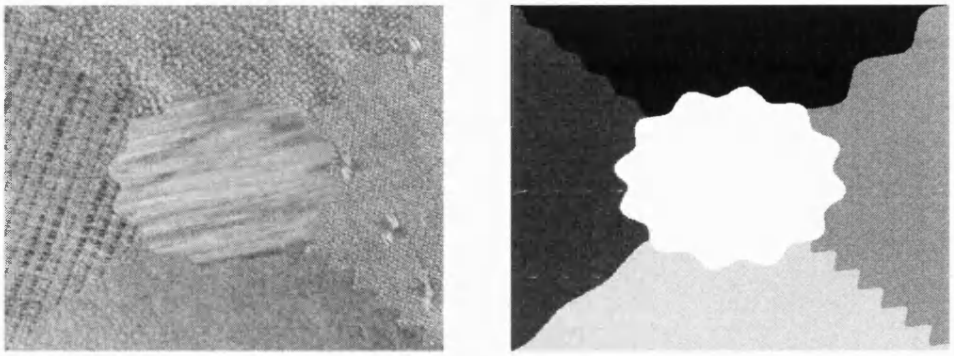
Figure 6.8: Unsupervised segmentation of compound Brodatz texture image utilising area pattern spectrum features with spatial coordinates.

an averaged confusion matrix from 3 segmentations of Outex images, which has an average classification accuracy of 93.78%.

The results presented in Tables 6.5 and 6.6 confirm area morphology local granulometries are a valid vehicle for texture feature extraction, therefore, they are suitable for the application of SAR image segmentation. For the segmentation of SAR imagery logarithmically spaced grainsizes were selected enabled the reduction of feature space by compounding larger scale features, which convey less textural information, whilst ensuring smaller scale features were captured independently. For a comparison of the effectiveness of area morphology based SAR segmentation with Gabor and wavelet based schemes, an unsupervised segmentation algorithm was developed with similarities to algorithms developed in Sections 4.3 and 5.4. After the multi-scale feature extraction stage no non-linear operator is required as the image transform is itself in-

True	Classified as:				
class:	D77	D84	D55	D17	D24
D77	99.94%	0.06%	0.00%	0.00%	0.00%
D84	5.45%	88.34%	0.00%	2.49%	3.73%
D55	5.81%	0.00%	89.35%	3.20%	1.64%
D17	0.00%	0.03%	2.03%	97.94%	0.00%
D24	4.97%	0.71%	1.38%	2.20%	90.74%

Table 6.5: Confusion matrix for unsupervised segmentation of compound Brodatz texture image utilising area pattern spectrum features with spatial coordinates.



(a) Outex sample segmentation problem. (b) Reference image.

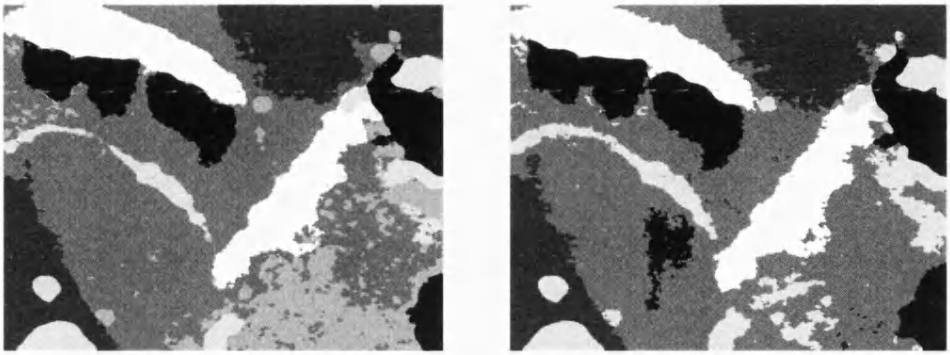
Figure 6.9: Samples from Outex database of texture segmentation images.

herently non-linear. However, for consistency with the Gabor and wavelet based segmentation results a Gaussian postfiler is applied to the multi-scale feature images, with results presented in Figure 6.10.

The application of a linear Gaussian postfilter to the grain images has the affect of removing the strong causality property in the feature extraction stage. This disadvantage can be overcome by utilising watershed segmentations to identify regions of homogeneous texture and adjusting an image features accordingly, effectively of preserving texture boundaries in the feature space. Segmentation results utilising watershed segmentation regions are shown in Figure 6.11.

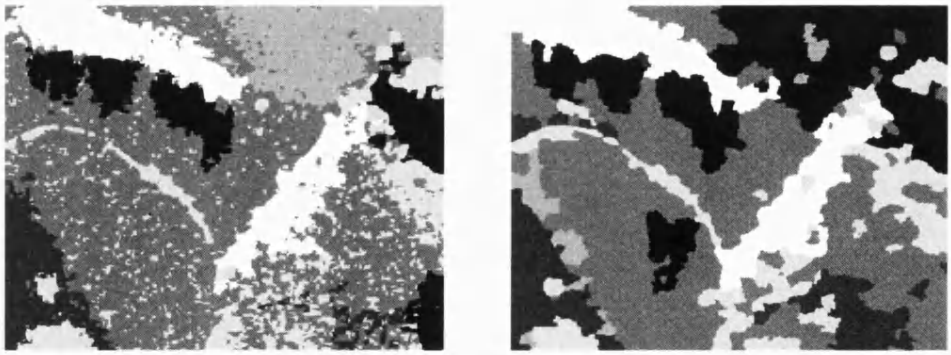
True	Classified as:				
class:	1	2	3	4	5
1	98.57%	0.65%	0.72%	0.00%	0.05%
2	1.58%	98.09%	0.00%	0.23%	0.10%
3	2.83%	0.00%	95.21%	1.50%	0.45%
4	0.00%	3.10%	3.55%	93.01%	0.35%
5	4.91%	3.73%	4.15%	3.19%	84.02%

Table 6.6: Average confusion matrix for unsupervised segmentation of Outex textures.



(a) Without speckle reduction. (b) With speckle reduction.

Figure 6.10: Granulometric unsupervised results.



(a) Without speckle reduction. (b) With speckle reduction.

Figure 6.11: Post segmentation granulometric unsupervised results.

6.6 Discussion

Comparison of the results in Tables 6.2 and 6.3 show an improved segmentation can be achieved by utilising greyscale pattern spectra. The overall correct classification, given by the mean value of the diagonal for each confusion matrix, rises from 77.18% (binary) to 83.15% (greyscale). However, many intra-region classification errors are still present resulting, in part, from the introduction of new features with increasing scale [2]. The results for the area morphology local granulometry segmentation, see Figure 6.7, are an improvement on the greyscale structuring element approach with an average correct classification of 89.95%. Finally area morphology local granulometries, when applied to the unsupervised segmentation of Figure 4.3(f), with spatial coordinates of each feature vector as additional features, gave an average correct classification result of 93.2620%.

When local area morphology granulometries are applied to extracting textural features from SAR imagery, the resulting segmentation did manage to distinguish between different regions within the image. However, when granulometric shrinkage was employed to remove speckle noise components from the small scale grains, this had the effect of vastly reducing the number of intra region classification errors. The most salient response to removing speckle noise components from the texture feature extraction framework is the region of shadow, localed near the centre of the image, is indistinguishable in the presence of speckle noise.

A disadvantage of using granulometries for textural feature extraction is a full decomposition results in a vast feature space *e.g.* for an image of size $N \times N$ pixels, it is possible to define a feature space with $2(N^2 - 1)$ dimensionality. One method of reducing the feature space to a more manageable size is to compound the grain images, a process similar to the binning of histogram data. However, this will introduce a quantisation error in the feature space and hence introduce uncertainty into the classification stage. An alternative to compounding the grain images is to reduce to feature space by identifying the salient scales which contain textural information, and only include these features. This makes the area morphology local granulometric texture segmentation scheme a data dependent algorithm and introduces a computational expensive pre-processing stage.

Chapter 7

Conclusions

THIS research has focused on the application of multi-scale image analysis techniques for the texture analysis of SAR imagery. Texture segmentation of remotely sensed data is important as textured regions of SAR images correspond to different ground cover types, providing a method for monitoring changes in the Earth's surface.

An understanding of texture analysis in SAR imagery is addressed in Chapter 1, which discusses two fundamental components, namely SAR image properties and image texture analysis. An insight into the properties of SAR imagery is obtained by examining the image formation process, which highlights the phenomenon of a noise like image component, namely *speckle*. A theoretical analysis of SAR image properties shows how it is possible to develop models to synthesise radar speckle, which have proved to be a very useful tool in later chapters. The second part of Chapter 1 provides a background to the area of texture analysis. The statistical properties of textures are studied including single point statistics, which can be found by analysis of the probability distribution function (pdf), and two point statistics contained in the grey-level co-occurrence matrices (GLCM). Chapter 1 is also describes how features can be derived from a textures statistical properties to proved texture discrimination, capable of representing an image in a feature domain. If textural information is available about an image prior to processing then supervised minimum distance classification algorithms are suitable for obtaining a segmentation. However, if no prior information is avail-

able about the texture present in an image, texture segmentation can be achieved via unsupervised clustering algorithms.

Chapter 2 is concerned with multi-scale image analysis techniques and details how principals are derived from models of the human visual system (HVS). Gabor's theory behind a joint spatial/spatial-frequency representation of signals and images is described as is the wavelet transform. The development a filterbank which is applicable to image processing is presented providing a means of multi-scale image decomposition. Traditionally the discrete wavelet transform is applied recursively to the low frequency subbands of an image, which is not optimum for the extraction of texture features. An alternative method of applying the wavelet transform, the wavelet packet transform, is studied which extracts information from all frequency subbands of an image, hence making it more suitable for the application of texture analysis. An alternative non-linear set of image analysis operators, namely mathematical morphology, are defined which analysis an image based on shape. By applying increasing scale morphological operators iteratively to an image a data sieve is constructed. The outputs of the data sieve are analogous with linear band-pass and low-pass filters, providing an alternative to linear filterbanks for multi-scale image analysis. By implementing a data sieve with area morphology operators, an image decomposition is produced, yielding information about all image features at all scales. From an area morphology decomposition rotation invariant texture features are obtained. Chapter 2 also studies the multi-scale segmentation of images via the watershed segmentation. With the use of markers, the watershed transform identifies varying scale regions of an image. The watershed transform was implemented in the post processing stage of feature extraction, leading to improved localisation of texture boundaries in the feature space.

The process of suppressing the speckle noise component in SAR imagery is addressed in Chapter 3. In particular this research has focused on speckle reduction algorithms that utilise multi-scale image analysis techniques. Multi-scale speckle suppression techniques are of interest as they have the potential of being incorporated in a multi-scale texture analysis framework. Through this investigation the novel granulometric shrinkage algorithm has been developed which has the property of preserving an images textural information. The final section of Chapter 3 are concerned with evaluating the effectiveness of algorithms at reducing speckle. In the evaluation of speckle reduction two test images are considered, one possessing single point noise, the other pos-

sessing correlated noise which is present in SAR imagery. Of the four speckle reduction techniques tested, namely non-redundant wavelet shrinkage; redundant wavelet shrinkage; area open-close; granulometric shrinkage; all were effective at reducing speckle noise in SAR imagery. However, a salient conclusion that can be drawn from analysis of the speckle reduction results is the morphology based speckle reduction algorithms possessed improved edge preservation when compared to the linear wavelet transform techniques.

The results in Chapter 4 show how texture features obtained via a Gabor filterbank are effective at segmenting an image comprising of Brodatz textures. However, only limited success of segmenting SAR imagery was achieved. This may be because a Gabor filterbank captures texture features on a scale that does not lend itself to high resolution SAR images.

Chapter 5 was concerned with the analysis of image texture utilising the wavelet transform. Both the non-redundant and redundant implementations of the wavelet transforms were studied, resulting in successful segmentation algorithms. The reasoning behind the suitability of the redundant wavelet transforms to texture feature extraction is the introduction of shift invariance. However, analysis of the segmentation results show only marginally improved segmentation results are achieved with the redundant wavelet transform. An advantage of the non-redundant wavelet transform is the reduction in space required for data storage which is a critical factor in spaceborne applications. Also the application of the non-redundant wavelet transform fits with image compression schemes.

Texture features obtained via area morphology operators are studied in Chapter 6. Existing morphological texture segmentation algorithms rely on the analysis of an image with multiple structuring elements. An advantage of area morphology texture features is the calculation of a single pattern spectrum is adequate for extracting texture features at all orientations, hence, the calculation of pattern spectrum moments as texture features is eliminated. Area morphology local granulometries have been shown to produce improved segmentation results in comparison to structuring element-based algorithms. This is due, in part, to the removal of artificial artefacts generated in the feature space with the use of structuring element-based morphological operators.

An advantage of segmenting an image with wavelet and morphology texture features is the ability to suppress speckle noise components in the feature space. This leads to a more robust segmentation, which is not possible with Gabor segmentation algorithms. Of all the segmentation algorithms tested it is felt the wavelet-based algorithms provide the best segmentation results. A quantitative measure can be obtained by evaluating the confusion matrices of the Brodatz segmentation results. Also a visual inspection of the SAR segmentation results show how regions in the wavelet segmentations correspond well to textured region in the test image.

7.1 Future work

Area morphology local granulometries are relatively new texture descriptors and have the potential for development. It is felt that improved segmentation could be achieved by including the mean value of a pixel's neighbourhood. One suitable method would be to include the coarsest scale sieved image as an extra feature, leading to less fragmented segmentation whilst preserving the property of edge localisation.

Another issue with area morphology local granulometries is the potential to establish a vast feature space. For an $N \times N$ pixel image a full decomposition produces a $2(N^2-1)$ dimension features space, with the inclusion of anti-granulometries, which is too large to cluster. One method of reducing the feature space is to histogram the grain responses of image pixels. However, quantising the feature space adds uncertainty in the classification stage and reduces the ability to discriminate between different textures. This problem can be overcome by introducing a preprocessing stage that identifies the most salient scales at which texture information is present. This would have the effect of removing redundant information from the feature space.

An area of current research is the inclusion of an adaptive threshold to the granulometric shrinkage process. By assessing if a pixel's local neighbourhood possesses textural information or is a uniform region, a threshold can be set on a per-pixel basis leading to a global optimum reduction in speckle whilst preserving image features.

Appendix A

Wavelet Filters

THIS appendix describes the design considerations concerning the two-channel wavelet filterbank in Figure A.1. Before a detailed description of the filterbank is given the fundamental operators need to be defined.

Given $A = \{h_0, h_1, \ldots, h_{L-1}\}$, filtering an input signal $x(n)$ means:

$$a(k) = \sum_{n=0}^{L-1} h(n)x(n-k) \tag{A.1}$$

Down-sampling means:

$$a_0, a_1, a_2, a_3, \ldots \Longrightarrow a_0, a_2, a_4, \ldots \tag{A.2}$$

Up-sampling means:

$$c_0, c_1, c_2, c_3, \ldots \Longrightarrow c_0, 0, c_1, 0, c_2, 0, \ldots \tag{A.3}$$

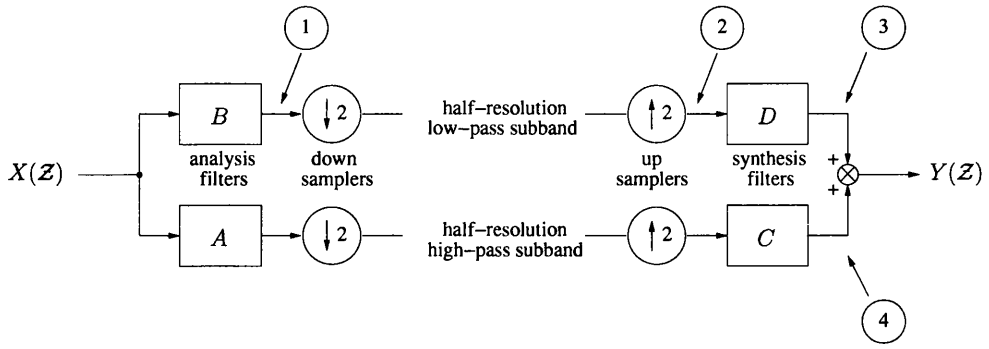


Figure A.1: Two-channel wavelet filter bank.

A.1 $\mathcal{Z}$ Transform

The $\mathcal{Z}$ transform was introduced to transform a discrete-time signal $x(n)$, to the complex $\mathcal{Z}$ plane [81]. A definition of the $\mathcal{Z}$ transform is given in (A.4). In the $\mathcal{Z}$ domain $\mathcal{Z}^{-1}$ is a time delay of 1 sample, and $\mathcal{Z} = exp\{j\omega T\}$ gives the frequency response of a system.

$$X(\mathcal{Z}) = \sum_{n=0}^{L-1} x(n)\mathcal{Z}^{-n} \tag{A.4}$$

Using the definition (A.4), the filtering process of (A.1) can be replaced by:

$$a(\mathcal{Z}) = A(\mathcal{Z})X(\mathcal{Z}) \tag{A.5}$$

The individual sampling operations in Figure A.2 can be expressed in the $\mathcal{Z}$ domain by considering the two operations together.

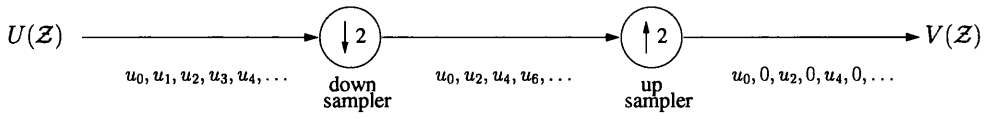


Figure A.2: Sampling operations.

From Figure A.2 the combined down and up sampling operators can be expressed

as [88]:

$$\begin{aligned} V(\mathcal{Z}) &= \sum_{n=0}^{L-1} u(n)\mathcal{Z}^{-n}\frac{1+(-1)^n}{2} \\ &= \tfrac{1}{2}\sum_{n=0}^{L-1} u(n)\mathcal{Z}^{-n} + (-1)^n\mathcal{Z}^{-n}u(n) \\ &= \tfrac{1}{2}\{U(\mathcal{Z}) + U(-\mathcal{Z})\} \end{aligned}$$

A.2 Wavelet Filterbank

Using the functions in Figure A.1 expressed in the $\mathcal{Z}$ domain, it is possible to calculate the output of the filterbank. This is achieved by considering the output at mid-points in the filterbank. At position ① the output is:

$$X(\mathcal{Z})B(\mathcal{Z})$$

After down and up sampling the signal at position ② is:

$$\frac{1}{2}\{X(\mathcal{Z})B(\mathcal{Z}) + X(-\mathcal{Z})B(-\mathcal{Z})\}$$

At position ③ the signal is:

$$\frac{1}{2}D(\mathcal{Z})\{X(\mathcal{Z})B(\mathcal{Z}) + X(-\mathcal{Z})B(-\mathcal{Z})\}$$

Similarly at ④:

$$\frac{1}{2}C(\mathcal{Z})\{X(\mathcal{Z})A(\mathcal{Z}) + X(-\mathcal{Z})A(-\mathcal{Z})\}$$

After synthesis the reconstructed signal is:

$$\begin{aligned} Y(\mathcal{Z}) &= \tfrac{1}{2}\{A(\mathcal{Z})C(\mathcal{Z}) + B(\mathcal{Z})D(\mathcal{Z})\}X(\mathcal{Z}) \\ &+ \tfrac{1}{2}\{A(-\mathcal{Z})C(\mathcal{Z}) + B(-\mathcal{Z})D(\mathcal{Z})\}X(-\mathcal{Z}) \end{aligned} \tag{A.6}$$

The output of the filterbank has two terms. The coefficient of the $X(\mathcal{Z})$ term introduces distortion, and hence is known as the *distortion term.* The term in $X(-\mathcal{Z})$ introduces unwanted frequency components and is known as the *aliasing term.*

A.2.1 Perfect Reconstruction

For perfect reconstruction the aliasing term must be eliminated, this is achieved by setting:

$$C(\mathcal{Z}) = B(-\mathcal{Z}) \Longrightarrow C(\mathcal{Z}) = B(\mathcal{Z}) \text{ with alterating signs} \tag{A.7}$$

$$D(\mathcal{Z}) = -A(-\mathcal{Z}) \Longrightarrow D(\mathcal{Z}) = -A(\mathcal{Z}) \text{ with alterating signs} \tag{A.8}$$

The output $Y(\mathcal{Z})$ in Figure A.1 is now:

$$Y(\mathcal{Z}) = \frac{1}{2}\{A(\mathcal{Z})B(-\mathcal{Z}) - B(\mathcal{Z})A(-\mathcal{Z})\}X(\mathcal{Z}) \tag{A.9}$$

For perfect reconstruction the distortion term must be reduced to a delay [122]:

$$A(\mathcal{Z})B(-\mathcal{Z}) - B(\mathcal{Z})A(-\mathcal{Z}) = 2\mathcal{Z}^{-k} \tag{A.10}$$

From (A.7) this gives:

$$A(\mathcal{Z})C(\mathcal{Z}) - A(-\mathcal{Z})C(-\mathcal{Z}) = 2\mathcal{Z}^{-k} \tag{A.11}$$

Let:

$$P(\mathcal{Z}) = A(\mathcal{Z})C(\mathcal{Z}) \tag{A.12}$$

To design a perfect reconstruction filterbank, a polynomial $P(\mathcal{Z})$ has to be found which satisfies (A.12). Once $P(\mathcal{Z})$ is expressed as 2 factors this identifies the filters $A(\mathcal{Z})$ and $C(\mathcal{Z})$ [88]. For the case when $C(\mathcal{Z})$ is a time reversal of $A(\mathcal{Z})$ the condition (A.11) is:

$$A(\mathcal{Z})A(\mathcal{Z}^{-1}) + A(-\mathcal{Z})A(-\mathcal{Z}^{-1}) = 2 \tag{A.13}$$

A.2.2 Regularity

For improved results filter $B(\mathcal{Z})$ in Figure A.1 should behave as a high pass filter [88]. This means if the input is constant the output should be zero *i.e.* the filter should block d.c. signals. For a filter to be high pass all the coefficients should sum to zero. From (A.13) the d.c. response can be calculated by setting $\mathcal{Z} = 1$:

$$\begin{aligned} A(1)A(1) + A(-1)A(-1) &= 2 \\ A(1)^2 + A(-1)^2 &= 2 \\ \left\{\sum_{n=0}^{L-1} h(n)1^n\right\}^2 + \left\{\sum_{n=0}^{L-1} h(n)(-1)^n\right\}^2 &= 2 \end{aligned}$$

As the sum of coefficients of a high pass filter sum to zeros this give the *regularity condition*:

$$\sum_{n=0}^{L-1} h(n) = \sqrt{2} \tag{A.14}$$

Appendix B

Wavelet Extensions

WHEN a signal of finite length is analysed via a two-channel wavelet filterbank in Figure A.1, the problem of handling the boundary conditions at the ends of the signal arises. It is important to handle the boundary conditions in a manner to preserve the perfect reconstruction property of the filterbank, without increasing the variance of the highpass filtered subband coefficients.

B.1 Periodic Extension Transform

For a finite signal $x(n)$, of length N_x, the traditional method is to apply the filters of Figure A.1 to the infinite periodic extension of $x(n)$. If the FIR filters are of length N_F, then for $N_f \leq N_x$, this equates to circular convolution of period N_x. The extension of the input signal, with a period equal to the length of the input vector,describes a linear transform for finite length signals known as the *periodic extension transform.*

The filter bank can be viewed as being fixed, whilst the extension is a linear preprocessing stage to the transform. This gives an understanding of how the extensions can be implemented when developing the algorithms. This view also allows flexibility, by keeping the filterbank fixed and defining a number of multi-dimensional linear transforms [21]. However, the periodic extension transform has a disadvantage by in-

troducing a transition where the ends of the signal meet. This discontinuity can result in an increase in variance of the high frequency subband coefficients.

B.2 Symmetric Extension Transform

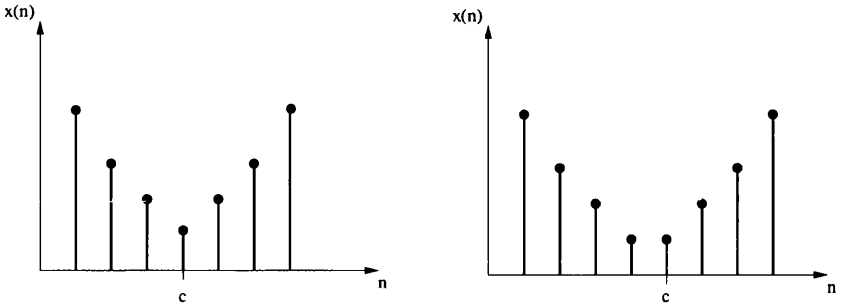
A popular image coding algorithm is to quantise the output of the two-dimensioal discrete cosine transform (DCT). The DCT can be obtained from the first half of the discrete Fourier transform (DFT) of the signal $y(n)$ (B.1). The signal $y(n)$ is of length $2N_x$ obtained by the symmetric extension, or reflection, of $X(n)$.

$$y(n) = \begin{cases} x(n); & n = 0, \ldots, N_x - 1 \\ x(2N_x - 1 - n; & n = N_x, \ldots, 2N_x - 1 \end{cases} \tag{B.1}$$

The fact that the extension $y(n)$ is an even function, results in the periodisation artefact being greatly reduced. Hence the efficiency of the image coding algorithm is increased. A similar technique can be implemented for transforming finite signals with a two-channel wavelet filterbank. For the input signal $x(n)$ of length N_x, the symmetric extension, $y(n)$ of length N_y can be generated. The extension signal can then be transformed by N_y periodic convolution, which eliminates the transition that results from the periodic extension of $x(n)$. After the output of the synthesis filterbank, the original signal $x(n)$ can be recovered by selecting N_x samples of the output signal.

Symmetric extension transforms have been developed for two-channel multi-rate filterbanks [21]. The described extensions allow for the non-redundant, perfect reconstruction transform of signals. Another advantage of symmetric extension transforms are they can be applied to input signals for filters of arbitrary lengths, both odd and even. A discrete time signal can be symmetric in two ways:

1. Symmetric about one of its samples, known as *whole-sample symmetry.*
2. Symmetric about a midway point between samples, known as *half-sample symmetry.*



(a) Whole sample symmetry (b) Half sample symmetry

Figure B.1: Examples of symmetric extensions.

Examples of the symmetric extensions are shown in Figure B.1. For whole-sample symmetry the signal is symmetric about the sample c, whereas for half-sample symmetry the signal is symmetric about $c - \frac{1}{2}$. In cases where the $N_x > N_f$, filters of odd lengths utilise whole-sample symmetry, and even length filters utilise half-sample symmetry.

Appendix C

Set Theory

SETS can be defined as a collection of objects or *elements*. A set can be described by listing the whole collection of elements enclosed in braces{ }. By convention an upper-case letter is used to represent a set, while a lower-case letter is used to represent an element of a set *e.g.* a set A could contain the set of digits used in the decimal system:

$$A = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\} \tag{C.1}$$

To state a particular element is a member of a particular set the symbol $\in$ is used, which means 'is a member of'. To assist with the description of set theory a list of symbols are used to represent some standard sets, see Table C.1, *e.g.* $2 \in A$ and $2 \in \mathbb{N}$. Similarly $\notin$ means 'is not member of', *e.g.* $-5 \notin A$.

Another way of describing a set is to define a rule by which all the elements of the set can be found, *e.g.* $A = \{a : a \in \mathbb{N} \text{ and } a \leq 9\}$. This reads '$A$ is the set of values a such that a is a member of non-negative integers and a is less than or equal to 9'.

$\mathbb{N}$	The set of non-negative integers
$\mathbb{N}^+$	The set of positive integers
$\mathbb{Z}$	The set of integers, positive, negative and zero
$\mathbb{R}$	The set of all real numbers
$\mathbb{R}^+$	The set of positive real numbers
$\mathbb{R}^-$	The set of negative real numbers
$\mathbb{C}$	The set of all complex numbers
$\mathbb{Q}$	The set of all rational numbers
$\mathbb{T}$	The set of all irrational numbers

Table C.1: Standard sets

C.1 Venn Diagrams

Venn diagrams provide a graphical method of visualising sets, this can easily be shown by use of an example. Let $X = \{4, 5, 6, 7\}$ and $Y = \{2, 4, 6, 8\}$. The set containing all the numbers, the *universal set* $= \mathbb{E}$, is defined in (C.1). Sets are drawn as regions (usually circles) and the universal set is represented by a rectangular region. An example of a Venn diagram is shown in Figure C.1, and from this representation various properties can be observed.

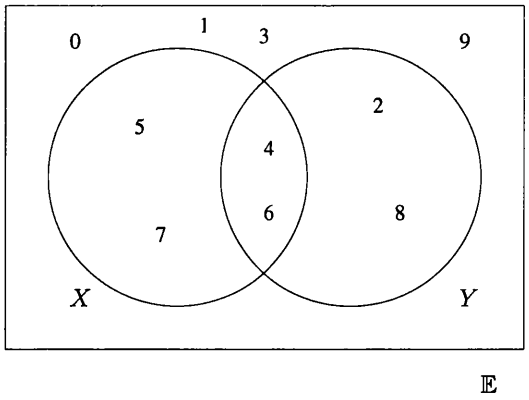


Figure C.1: An example Venn diagram.

The sets X and Y are represented by the interiors of the circles. Properties that can be observed from Figure C.1 are, $5 \in X$, $8 \in Y$, $9 \notin X$ and $9 \notin Y$. It is also clear that

elements 4 and 6 are common to both sets.

C.2 Intersection

A useful property of sets is to know the common elements, and leads to the definition of *intersection.* Given two sets A and B, a new set containing the elements common to both A and B can be written as:

$$A \cap B = \{x : x \in A \text{ and } x \in B\} \tag{C.2}$$

In the Venn diagram example in Section C.1, it can be observed that $X \cap Y = \{4, 6\}$, that is, $4 \in X \cap Y$ and $6 \in X \cap Y$. If the set $A \cap B$ contains no elements, this implies A and B are *disjoint* and write $A \cap B = \emptyset$, where $\emptyset$ denotes the *empty set.*

C.3 Union

Given two sets A and B, we can define a set which contains all the elements of both sets. This set is called the *union* of sets A and B, and can be written as:

$$A \cup B = \{x : x \in A \text{ or } x \in B \text{ or both}\} \tag{C.3}$$

In the Venn diagram example in Section C.1, it can be observed that $X \cup Y = \{2, 4, 5, 6, 7, 8\}$. The elements $X \cap Y$ are only required to be listed once.

C.4 Subsets

It is possible for a set to be contained within another set. If all the elements of a set A are also elements of a set B, then set A is a *subset* of B, and can be written as $A \subset B$. From Table C.1 it can be seen that $\mathbb{N} \subset \mathbb{Z}$ and $\mathbb{Z} \subset \mathbb{R}$. If a set C fits exactly inside another set D, with all the elements common to both sets then $C \subseteq D$, and neither set cantains elements which are not present in the other set. An effective way of visualising this property is to use a Venn diagram, Section C.1. For the sets:
$X = \{x : x \in \mathbb{N}^+ \text{ and } x \leq 9\}$;
$Y = \{x : x \in \mathbb{N}^+ \text{ and } x < 10 \text{ and } x \text{ is odd}\}$;
$Z = \{x : x \in \mathbb{N}^+ \text{ and } x \leq 7 \text{ and } x \text{ is odd}\}$;
given the universal set $\mathbb{E}$ defined in (C.1) the corresponding Venn diagram is shown in Figure C.2.

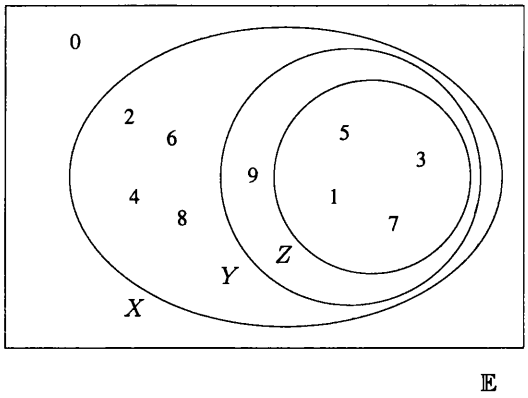


Figure C.2: Venn diagram illustrating subsets.

Properties observed from Figure C.2 are:

$Y \subset X$ Y is contained within set X.
$Z \subset Y$ Z is contained within set Y.
$Z \subset X$ Z is contained within set X.
$Z \subseteq Z$ Z is contained within set Z with all elements common to both sets.

C.5 Complement

For a pre-defined universal set $\mathbb{E}$ and set A, where $A \in \mathbb{E}$, the set of elements not in A are called the *complement* of A and is represented by $\overline{A}$. From the previous set operators it can be derived that $A \cap \overline{A} = \emptyset$ and $A \cup \overline{A} = \mathbb{E}$.

C.6 Set Algebra

From the operations in the previous sections it is possible to develop the laws of set algebra, see Table C.2. According to these rules it is possible to simplify expressions involving set operations. Using the laws in Table C.2 it is possible to derive the laws in Table C.3.

$A \cup B = B \cup A$ $A \cap B = B \cap A$	Commutative laws
$A \cup (B \cup C) = (A \cup B) \cup C$ $A \cap (B \cap C) = (A \cap B) \cap C$	Associative laws
$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$ $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$	Distributive laws
$A \cup \emptyset = A$ $A \cap \mathbb{E} = A$	Identity laws
$A \cup \overline{A} = \mathbb{E}$ $A \cap \overline{A} = \emptyset$ $\overline{\overline{A}} = A$	Complement laws
$A \cup A = A$ $A \cap A = A$	Idempotent laws

Table C.2: Laws of set algebra

C.7 Sets and Functions

Sometimes it is essential to investigate relationship between the elements of sets. For a set $A = \{0, 1, 2, 3, 4\}$ and a set $B = \{1, 2, 4, 8, 16\}$, it can be seen each element of

$A \cup (A \cap B) = A$ $A \cap (A \cup B) = A$	Absorption laws
$(A \cap B) \cup (A \cap \overline{B}) = A$ $(A \cup B) \cap (A \cup \overline{B}) = A$	Minimisation laws
$\overline{A \cup B} = \overline{A} \cap \overline{B}$ $\overline{A \cap B} = \overline{A} \cup \overline{B}$	De Morgan's laws

Table C.3: Laws derived from Table C.2

B is 2 raised to the power of an element in A, and is visualised in Figure C.3.The rule which produces an element in B, when given an element in A is known as a *relation.* A relation r can be written as $r : A \rightarrow B$, which means 'the relation r maps the elements of set A to the elements of set B'. For the example in Figure C.3, $r : 0 \rightarrow 1, r : 1 \rightarrow 2$ or in general form $r : x \rightarrow 2^x$. The set which is chosen as the input is called the *domain*, and the set which it is mapped to is called the *co-domain.*

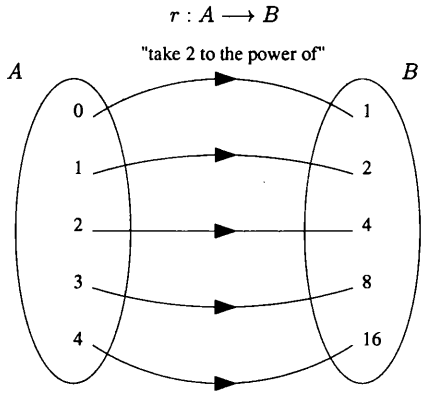


Figure C.3: A relationship between two sets.

Appendix D

Basic Morphological Operators and Properties

THE aim of morphological operations is to extract structures from an signal using its subgraph representation. This is achieved by examining the signal with a set of known shape called a *structuring element* (SE). The geometric shape of the SE is usually based on *a priori* knowlege of structures of interest. The SE can also be based on irrelevant structures, which are objects that are required to be removed.

D.1 Binary Morphology

The two fundamental morphological operators are *dilation* and *erosion*, and can be used as the building blocks to create more complex operations. For detailed analysis of these operators the reader is refered to [59]. For two sets $X, B \subseteq \mathbb{E}^d$, dilation can be defined by the *Minkowski addition* (D.1).

$$X \oplus B = \{x + b | x \in X, b \in B\} \tag{D.1}$$

If a structuring element is chosen $B \subseteq \mathbb{E}^d$, the dilation of a set X by B is defined as:

$$\delta_B(X) = X \oplus B \tag{D.2}$$

Dilation can be defined in terms of a union of set translations (D.3), the translations being defined by the SE, where $\check{B} = \{-b | b \in B\}$ is a reflection of B with respect to the origin. For a 1-dimensional signal an example of binary dilation is given in D.2(c).

$$\delta_B(X) = \{y : \check{B}_y \cap X \neq \emptyset\} = \bigcup_{b \in B} X_b \tag{D.3}$$

The set B can replace $\check{B}$ if a symmetrical SE is used. Properties of the dilation operator are:

Translation invariant: $(X \oplus B)_t = X_t \oplus B$

The dilation X with B translated will produce the same result as the dilation of X translated with B.

Order invariant: $(X \oplus B) \oplus C = X \oplus (B \oplus C)$

If several dilation operations are to calculated, the order in which they are applied is irrelevant. This provides a method to decompose structuring elements into smaller parts.

Figure D.1: Decomposition of a structuring element.

Increasing operator: $X \subseteq Y \Rightarrow X \oplus B \subseteq Y \oplus B$

An operator r is defined as increasing if for two sets, A which is a subset of B, then $r(A)$ is still a subset of $r(B)$.

Scale invariant: $sX \oplus sB = s(X \oplus B)$

If the signal, X, and SE, B, are multiplied by a scaling factor s, this produces the same result as scaling the dilation of X and B by s.

The operation dual to dilation is *erosion.*

$$\epsilon_B(X) = X \ominus B = \overline{(\overline{X} \oplus B)} \tag{D.4}$$

This can be proved by the *adjunction relation.* For two sets $\mathcal{L}$ and $\mathcal{M}$ we can define the operators $\epsilon : \mathcal{L} \to \mathcal{M}$ and $\delta : \mathcal{M} \to \mathcal{L}$. The pair (ϵ, δ) is an adjunction between $\mathcal{L}$ and $\mathcal{M}$ if:

$$\delta(Y) \subseteq X \Longleftrightarrow Y \subseteq \epsilon(X) \qquad \text{where } X \in \mathcal{L} \text{ and } Y \in \mathcal{M} \tag{D.5}$$

Substituting (D.4) in (D.3) defines erosion [82], which can be expressed as an intersection of set translations (D.6), where $B_y = \{b + y | b \in B\}$ is the translation of the set B along the vector y.

$$\epsilon_B(X) = \{y : B_y \subseteq X\} = \bigcap_{b \in B} X_{-b} \tag{D.6}$$

Properties of the erosion operator are the same as listed for dilation. For a 1-dimensional signal an example of binary dilation is given in D.2(d).

Dilations and erosions are the fundamental building blocks to construct more morphological operations. Although the erosion of a signal removes all structures that cannot contain the SE, one of the problems is it also reduces other structures [124], this is shown in the example in Figure D.2(d). Due to the non-linear properties of the erosion operator, there is no inverse operator to reconstruct the original signal. The process of trying to recover structures suppressed by the erosion operator leads to the morphological *opening* operator. The principal behind the opening operator is a dilation of the signal, previously eroded, using the same structuring element. Morphological opening is defined in (D.7), and an example is presented in Figure D.2(e).

$$X \circ B = (X \ominus B) \oplus B \tag{D.7}$$

Alternatively it may be required recover inital structures in a signal that has been dilated. This is leads to the morphological *closing* operator, and is achieved by eroding the dilated signal. Morphological opening is defined in (D.8), and an example is presented in Figure D.2(f).

$$X \bullet B = (X \oplus B) \ominus B \tag{D.8}$$

The opening and closing operators also posses properties the make them attractive in the field of signal processing. The first property is that openings and closings are dual transforms with respect to the set complement:

$$X \circ B = \overline{\overline{X} \bullet B} \tag{D.9}$$

This can be proved by using (D.8) and (D.4):

$$\begin{aligned} X \circ B &= \overline{[\overline{X} \oplus B] \ominus B} \\ &= \overline{[\overline{X \ominus B}] \ominus B} \\ &= \overline{\overline{[X \ominus B]} \oplus B} \\ &= X \circ B \end{aligned}$$

Simular to dilation and erosion, opening and closing are both increasing operators:

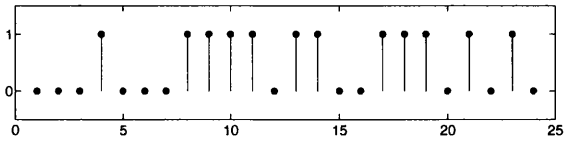
$$X \subseteq Y \Rightarrow \begin{cases} X \circ B \subseteq Y \circ B \\ X \bullet B \subseteq Y \bullet B \end{cases} \tag{D.10}$$

Another property of the opening and closing operators is that they are idempotent. The idempotence of a filter implies that a signal will bot be further modified by iterating the transform.

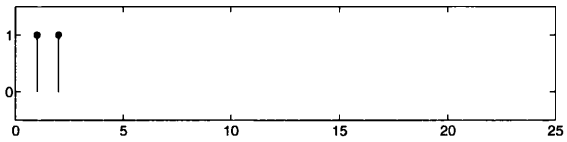
$$\begin{aligned} (X \circ B) \circ B &= X \circ B \\ (X \bullet B) \bullet B &= X \bullet B \end{aligned} \tag{D.11}$$

Simularly to dilations and erosions, openings and closings can be combined to generate more complex morphological operators. An opening followed by a closing (D.12), is used in granulometric applications in Section 2.5.1.

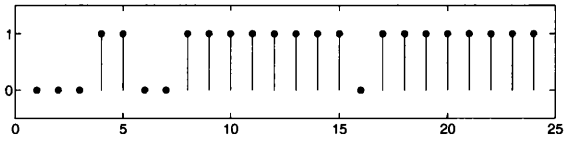
$$X \odot B = (X \circ B) \bullet B \tag{D.12}$$



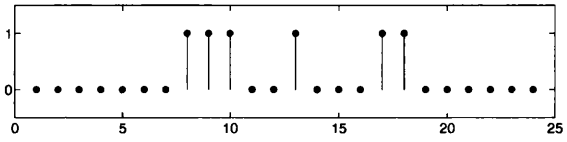
(a) Original signal



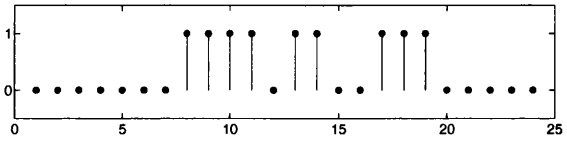
(b) Structuring element



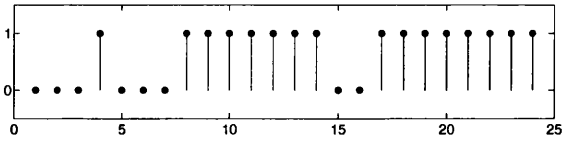
(c) Dilation



(d) Erotion



(e) Opening



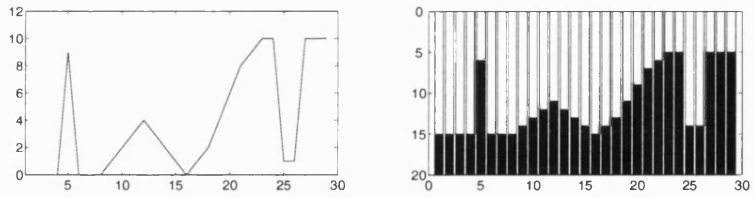
(f) Closing

Figure D.2: An example of binary morphological operations.

D.2 Grey-Scale Morphology

The operators explained in the previous section can be extended to two-dimensions and used in the area of images processing. In two-dimensions a signal becomes an image with horizontal and vertical axes. For processing images the SE can be a two-dimensional mask, or sub-image, which defines the pixels to be used in the morphological operations. The morphological operators previously defined have been used in two-dimensions for noise removal in binary images [108].

Since binary images make up a small subset of images, to make morphology more practical to the application of image processing the morphological operations were adapted for the use on grey-scale images [128] [57] [13]. The adaption of morphological operators from binary to grey-scale signals can be discribed with the example in Figure D.3. An alternative method of representing the 1-dimensional grey-scale signal in Figure D.3(a) is to express the signal as a 2-dimensional binary image. In the binary image D.3(b) all the points on or under the function are set to the one, and all the points above are set to zero. The grey-scale signal now casts a shadow, or *umbra*, below the function. The morphological processing of images can be discribed in an analogous way, where grey-scale images are represented by a 3-dimensional binary volume.



(a) Grey-scale signal (b) Binary image

Figure D.3: Grey-scale morphology.

The binary morphology operations of dilation, erosion, opening and closing can be extend to grey-scale images by the substitution of min and max for the $\cap$ and $\cup$ operations. Hence the dilated value of a pixel in an image I, at position (x, y), is the

maximum value in the sub-image defined by the SE B.

$$I \oplus B = \max_{(i,j)\in B} \{I(x+i, y+j)\} \tag{D.13}$$

Simularly the erosion of a grey-scale image can be defined as:

$$I \ominus B = \min_{(i,j)\in B} \{I(x+i, y+j)\} \tag{D.14}$$

Appendix E

Chi-Square Test

GIVEN data sets, it can be an important task to assess whether the two populations are from the same probability distrubution function. This can be expressed in statistical language [101]:

> Can we disprove, to a certain required level of significance, the null hypothesis that two data sets are drawn from the same population distribution function.

The task of proving that two discrete distributions come from the same population, or that they are different, can be evaluated using the *chi-Square test*. If an observed binned data set, P_o, is to be compared to an expected distribution, P_e, a measure of discrepancy can be calculated with (E.1). The summation is over all bins of the histogram, with any term $P_o(n) = P_e(n) = 0$ being ommitted from the sum.

$$\chi^2 = \sum_{n=1}^{N} \frac{(P_o(n) - P_e(n))^2}{P_e(n)} \tag{E.1}$$

When comparing two binned data sets, $P_1(n)$ and $P_2(n)$, then (E.2) should be used to obtain a measure of discrepancy.

$$\chi^2 = \sum_{n=1}^{N} \frac{(P_1(n) - P_2(n))^2}{P_1(n) + P_2(n)} \tag{E.2}$$

A large value of χ^2 indicates that the null hypothesis is rather unlikely. Another method of evaluation is the *chi-square probability function* $P(\chi^2, d)$ which is an incomplete Gamma distribution (E.3) [101]. The chi-square probability function is the probability that the sum of the squares of d random variables will be greater than χ^2, where the number of degrees of freedom, d, is taken as $N - 1$.

$$P(\chi^2, d) = \frac{1}{\Gamma(d)} \int_0^{\chi^2} \exp(-t) t^{(d-1)} dt \tag{E.3}$$

Appendix F

Author's Publications

N. D. Fletcher and A. N. Evans, "Minimum distance texture classification of SAR images using wavelet packets," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 02)*, vol. 3, pp. 1438–1440, 2002.

N. D. Fletcher and A. N. Evans, "Texture preserving speckle reduction in SAR imagery using granulometric shrinkage," in *Proc. of the IEE International Conference on Visual Information Engineering (VIE 05)*, pp. 203–210, 2005.

N. D. Fletcher and A. N. Evans, "Texture segmentation using area morphology local granulometries," in *Proc. of the International Symposium on Mathematical Morphology: 40 Years On (ISMM 05)*, pp. 367–376, 2005.

References

[1] K. W. Abyoto, S. J. Wirdjosoedirdjo, and T. Watanabe, "Unsupervised texture segmentation unsing multiresolution analysis for feature extraction," Tokyo University of Information Sciences, Tech. Rep., 1998.

[2] S. T. Acton and D. P. Mukherjee, "Scale space classification using area morphology," *IEEE Transactions on Image Processing*, vol. 9, no. 4, pp. 623–635, 2000.

[3] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Transactions on Image Processing*, vol. 1, no. 2, pp. 205–220, 1992.

[4] G. V. April and E. R. Harvey, "Speckle statistics in four-look synthetic-aperture radar imagery," *Optical Engineering*, vol. 30, no. 4, pp. 375–380, 1991.

[5] G. R. Arce and R. L. Stevenson, "On the synthesis of median filter systems," *IEEE Transactions on Circuits and Systems*, vol. 34, no. 4, pp. 420–429, 1987.

[6] N. S. Arini, "Post-segmentation feature-based classification of synthetic aperture radar data," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 02)*, vol. 3, pp. 1615–1617, 2002.

[7] ——, "Feature based classification of SAR clutter," DERA, Malvern, UK, Tech. Rep. DERA/S&E/RAD/WP001517/0.1, 2000.

[8] H. H. Arsenault and G. April, "Properties of speckle integrated with a finite aperture and logarithmically transformed," *Journal of the Optical Society of America*, vol. 66, no. 11, pp. 1160–1163, 1976.

[9] E. Avianto and M. Ito, "Speckle reduction for ultrasonic images using fuzzy morphology," *IEICE Transactions on Information & Systems*, vol. E84-D, no. 4, pp. 502–510, 2001.

[10] S. Baeg, A. T. Popov, V. G. Kamat, S. Batman, K. Sivakumar, N. Kehtarnavaz, E. R. Dougherty, and R. B. Shah, "Segmentation of mammograms into distinct morphological texture regions," in *Proc. 11th IEEE Symposium on Computer Based Medical Systems*, pp. 20–25, 1998.

[11] J. A. Bangham and T. G. Campbell, "Sieves and wavelets: multiscale transforms for pattern recognition," in *IEEE Winter Workshop on Nonlinear Digital Signal Processing*, pp. 1.1_4.1–1.1_4.6, 1993.

[12] J. A. Bangham, P. Chardaire, C. J. Pye, and P. D. Ling, "Multiscale nonlinear decomposition: The sieve decomposition theorem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 5, pp. 529–539, 1996.

[13] J. A. Bangham and S. Marshall, "Image and signal processing with mathematical morphology," *IEE Electronics & Communication Engineering Journal*, vol. 10, no. 3, pp. 117–128, 1998.

[14] A. Baraldi and F. Parmiggiani, "An investigation of the textural characteristics associated with gray level cooccurrence matrix statistical parameters," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 33, no. 2, pp. 293–304, 1995.

[15] A. Barni, M. Betti, and A. Mecocci, "Fuzzy segmentation of SAR images for oil spill recognition," *Proc. IEE International Conference on Image Processing and its Applications*, no. 410, pp. 534–538, 1995.

[16] S. Beucher, "The watershed transformation applied to image segmentation," *Scanning Microscopy International*, pp. 299–314, 1992.

[17] K. W. Bjerde, A. H. S. Solberg, and R. Solberg, "Oil spill detection in SAR imagery," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 93)*, vol. 3, pp. 943–945, 1993.

[18] D. Blacknell, "New method for the simulation of correlated k-distributed clutter," *IEE Proc. Radar, Sonar and Navigation*, vol. 141, no. 1, pp. 53–58, 1994.

[19] A. C. Bovik, "Analysis of multichannel narrow-band filters for image texture segmentation," *IEEE Transactions on Signal Processing*, vol. 39, no. 9, pp. 2025–2043, 1991.

[20] E. J. Breen and R. Jones, "Attribute openings, thinnings, and granulometries," *Computer Vision and Image Understanding*, vol. 64, no. 3, pp. 377–389, 1996.

[21] C. M. Brislawn, "Classification of nonexpansive symmetric extension transforms for multirate filter banks," Los Alamos National Laboratory, Los Alamos, NM, Tech. Rep., March 1996.

[22] P. Brodatz, *A Photographic Album for Artists and Designers.* New York: Dover, 1966.

[23] T. Chang and C. C. J. Kuo, "Tree-structured wavelet transform for textured image segmentation," in *Proc. of SPIE - the International Society for Optical Engineering*, vol. 1770, pp. 394–405, 1992.

[24] ——, "Texture analysis and classification with tree-structured wavelet transform," *IEEE Transactions on Image Processing*, vol. 2, no. 4, pp. 1376–1387, 1993.

[25] Y. Chen and E. R. Dougherty, "Gray-scale morphological granulometric texture classification," *Optical Engineering*, vol. 33, no. 8, pp. 2713–2723, 1994.

[26] F. Cheng and A. N. Venetsanopoulos, "An adaptive morphological filter for image processing," *IEEE Transactions on Image Processing*, vol. 1, no. 4, pp. 533–539, 1992.

[27] W. Chumsamrong, P. Thitimajshima, and Y. Rangsanseri, "Wavelet-based texture analysis for SAR image classification," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 99)*, vol. 3, pp. 1564–1566, 1999.

[28] J. G. Daugman, "Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters," *Journal of the Optical Society of America A-Optics & Image Science*, vol. 2, no. 7, pp. 1160–1169, 1985.

[29] Y. Dong, A. K. Milne, and B. C. Forster, "Toward edge sharpening: A SAR speckle filtering algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 39, no. 4, pp. 851–863, 2001.

[30] ——, "A review of sar speckle filters: texture restoration and preservation," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 2000)*, vol. 2, pp. 633–635, 2000.

[31] D. L. Donoho, "De-noising by soft-thresholding," *IEEE Transactions on Information Theory*, vol. 41, no. 3, pp. 613–627, 1995.

[32] E. R. Dougherty, J. B. Pelz, F. Sand, and A. Lent, "Morphological image segmentation by local granulometric size distributions," *Journal of Electronic Imaging*, vol. 1, no. 1, pp. 46–60, 1992.

[33] E. R. Dougherty, E. J. Kraus, and J. B. Pelz, "Image segmentation by local morphological granulometries," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 89)*, vol. 8, pp. 1220–1223, 1989.

[34] E. R. Dougherty, J. T. Newell, and J. B. Pelz, "Morphological texture-based maximum-likelihood pixel classification based on local granulometric moments," *Pattern Recognition*, vol. 25, no. 10, pp. 1181–1198, 1992.

[35] E. R. Dougherty and J. B. Pelz, "Pixel classification by morphologically derived texture features," in *Proc. of the SPIE Visual Communications and Image Processing*, vol. 1199, pp. 440–449, 1989.

[36] J. M. Durand, B. J. Gimonet, and J. R. Perbos, "SAR data filtering for classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 25, no. 5, pp. 629–637, 1987.

[37] L. V. Dutra, R. Huber, and P. Hernandez, "Primary forest and land cover contextual classification using JERS-1 data in Amazonia, Brazil," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 98)*, vol. 5, pp. 2743–2745, 1998.

[38] A. N. Evans, "A gamma filter for multi-look synthetic aperture radar images," in *International Symposium on Signal Processing and its Applications (ISSPA)*, pp. 829–832, 1996.

[39] N. D. Fletcher and A. N. Evans, "Minimum distance texture classification of SAR images using wavelet packets," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 02)*, vol. 3, pp. 1438–1440, 2002.

[40] ——, "Texture preserving speckle reduction in SAR imagery using granulometric shrinkage," in *Proc. of the IEE International Conference on Visual Information Engineering (VIE 05)*, pp. 203–210, 2005.

[41] ——, "Texture segmentation using area morphology local granulometries," in *Proc. of the International Symposium on Mathematical Morphology: 40 Years On (ISMM 05)*, pp. 367–376, 2005.

[42] L. Floreby, F. Sattar, and G. Salomonsson, "Image enhancement by morphological pyramid decomposition and modified reconstruction," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 97)*, vol. 4, pp. 2585–2588, 1997.

[43] A. C. Frery, C. da Costa Freitas Yanasse, and S. J. S. Sant'Anna, "Alternative distributions for the multiplicative model in SAR images," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 95)*, vol. 1, pp. 169–171, 1995.

[44] ——, "Statistical characterization of SAR data: The multiplicative model and extensions," in *Simposio Latinoamericano de Percepcion Remota (SELPER)*, Mexico, pp. 502–515, 1995.

[45] S. Fukuda and H. Hirosawa, "Multiresolution analysis and processing of synthetic aperture radar images using wavelets," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 97)*, vol. 3, pp. 1187–1189, 1997.

[46] D. Gabor, "Theory of communication," *Journal of the Institution of Electrical Engineers*, vol. 93, pp. 429–457, 1946.

[47] J. M. Gauch and S. M. Pizer, "Multiresolution analysis of ridges and valleys in grey-scale images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 6, pp. 635–646, 1993.

[48] T. Géraud, "Fast road network extraction in satellite images using mathematical morphology and Markov random fields," in *Proc. IEEE Workshop on Nonlinear Signal and Image Processing (NSIP 2003)*, Grando-Trieste, Italy, 2003.

[49] A. Gersho, "On the structure of vector quantizers," *IEEE Transactions on Information Theory*, vol. 28, no. 2, pp. 157–166, 1982.

[50] D. D. Giusto, L. Boroczky, R. Fioavanti, and S. Fioavanti, "SAR image filtering using wavelet transform," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 95)*, vol. 4, pp. 2153–2156, 1995.

[51] R. C. Gonzalez and R. E. Woods, *Digital Image Processing: 2nd edition.* Upper Saddle River, N.J.: Prentice Hall, 2002.

[52] J. W. Goodman, "Some fundamemtal properties of speckle," *Journal of the Optical Society of America*, vol. 66, no. 11, pp. 1145–1149, 1976.

[53] M. Grimaud, "A new measure of contrast: the dynamics," in *SPIE Image Algebra and Morphological Image Processing III*, vol. 1769, pp. 292–305, 1992.

[54] H. Guo, J. E. Odegard, M. Lang, R. A. Gopinath, I. W. Selesnick, and C. S. Burrus, "Wavelet based speckle reduction with application to SAR based ATD/R," in *Proc. IEEE International Conference on Image Processing (ICIP 94)*, vol. 1, pp. 75–79, 1994.

[55] D. J. Hand, *Lecture Notes: Probability and Statistics for Students of Computing*, Imperial College, London, UK, 2003.

[56] R. M. Haralick, K. Shanmugam, and I. Dinstein, "Textural features for image classification," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 3, no. 6, pp. 610–621, 1973.

[57] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 4, pp. 532–550, 1987.

[58] N. R. Harvey and S. Marshall, "Video and film restoration using mathematical morphology," in *IEE Colloquium on Non-Linear Signal and Image Processing*, pp. 3/1–3/5, 1998.

[59] H. J. A. M. Heijmans, "Mathematical morphology: basic principles," in *Proc. of Summer School on Morphological Image and Signal Processing*, Zakopane, Poland, pp. 1–18, 1995.

[60] Q. A. Holmes, D. R. Nüesch, and R. A. Shuchman, "Textural analysis and real-time classification of sea-ice types using digital SAR data," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 22, no. 2, pp. 113–120, 1984.

[61] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," in *Proc. IEEE International Conference on Systems, Man and Cybernetics*, pp. 14–19, 1990.

[62] ——, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[63] X.-Y. Jiang and R.-C. Zhao, "Texture segmentation based on incomplete wavelet packet frame," in *Proc. of the 2nd International Conference on Machine Learning and Cybernetics (ICMLC 2003)*, Xi-an, China, pp. 3172–3177, 2003.

[64] C. Ju and C. Moloney, "An edge-enhanced segmentation method for SAR images," in *Proc. Canadian Conference on Electrical and Computer Engineering (CCECE 97)*, vol. 2, pp. 599–602, 1997.

[65] A. Kher and S. Mitra, "An efficient speckle removing algorithm based on new morphological operators," in *Proc. 37th IEEE Midwest Symposium on Circuits & Systems*, vol. 2, pp. 852–855, 1994.

[66] T. Kohonen, "The self-organizing map," *Proc. of the IEEE*, vol. 78, no. 9, pp. 1464–1480, 1990.

[67] C. Kotropoulos and I. Pitas, "Optimum nonlinear signal detection and estimation in the presence of ultrasonic speckle," *Ultrasonic Imaging*, vol. 14, pp. 249–275, 1992.

[68] D. T. Kuan, A. A. Sawchuk, T. C. Strand, and P. Chavel, "Adaptive restoration of images with speckle," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 35, no. 3, pp. 373–383, 1987.

[69] A. Laine and J. Fan, "Texture classification by wavelet packet signatures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 11, pp. 1186–1191, 1993.

[70] M. Lang, H. Guo, J. E. Odegard, C. S. Burrus, and R. O. Wells, "Nonlinear processing of a shift invariant DWT for noise reduction," in *Proceedings of SPIE - The International Society for Optical Engineering*, vol. 2491/1, pp. 640–651, 1995.

[71] D. L. Lau, G. R. Arce, and N. C. Gallagher, "Robust image wavelet shrinkage for denoising," in *Proc. IEEE International Conference on Image Processing (ICIP 96)*, pp. 371–374, 1996.

[72] J.-S. Lee, "Digital image smoothing and the sigma filter," *Computer Vision, Graphics and Image Processing*, vol. 24, pp. 255–269, 1983.

[73] ——, "A simple speckle smoothing algorithm for synthetic aperture radar images," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 13, no. 1, pp. 85–89, 1983.

[74] W. Li, B. Bénié, D.-C. He, S. Wang, D. Ziou, and Q. H. J. Gwyn, "Classification of SAR images using morphological texture features," *International Journal of Remote Sensing*, vol. 19, no. 17, pp. 3399–3410, 1998.

[75] J. Liang and T. W. Parks, "A two-dimensional translation invariant wavelet representation and its applications," in *Proc. IEEE International Conference on Image Processing (ICIP 94)*, vol. 1, pp. 66–70, 1994.

[76] S. C. Liew, H. Lim, L. K. Kwoh, and G. K. Tay, "Texture analysis of SAR images," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS)*, vol. 2, pp. 1412–1414, 1995.

[77] Y.-C. Lin, T. Chang, and C. C. J. Kuo, "Texture segmentation using wavelet packets," in *Proc. of SPIE - the International Society for Optical Engineering*, vol. 2034, pp. 277–287, 1993.

[78] J. Malik and P. Perona, "Preattentive texture discrimination with early vision mechanisms," *Journal of the Optical Society of America A-Optics & Image Science*, vol. 7, no. 5, pp. 923–932, 1990.

[79] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[80] P. Maragos, "Pattern spectrum and multiscale shape representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 701–716, 1989.

[81] J. D. Martin, *Signals & Processes: A Foundation Course.* London: Pitman, 1991.

[82] G. Matheron, *Random Sets and Integral Geometrey.* New York: John Wiley and Sons, 1975.

[83] G. Matheron and J. Serra, "The birth of mathematical morphology," in *International Symposium on Mathematical Morphology (ISMM 02)*, pp. 1–16, 2002.

[84] J. McKenzie, S. Marshall, A. J. Gray, and E. R. Dougherty, "Parallel evolution functions for texture classification," in *Proc IEEE Workshop on Nonlinear Signal and Image Processing (NSIP 2003)*, Grando-Trieste, Italy, 2003.

[85] A. Meijster and M. H. F. Wilkinson, "A comparison of algorithms for connected set openings and closings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 484–494, 2002.

[86] Y. Meyer and R. D. Ryan, *Wavelets: Algorithms and Applications.* Philadelphia, PA: Society for Industrial and Applied Mathematics, 1993.

[87] Z. Mingjie, L. Xianfeng, and Y. Ruliang, "Speckle reduction for SAR images based on soft-morphological filter with omni-directional multiple structuring elements," in *Proc. CIE International Conference on Radar*, pp. 928–932, 2001.

[88] D. M. Monro, *Lecture Notes: Digital Video and Audio Course*, University of Bath, Bath, UK, 1998.

[89] P. V. Narasimha, M. S. R. R. Vidyadhar, T. C. M. Rao, and L. Venkataratnam, "An adaptive filter for speckle suppression in synthetic aperture radar images," *International Journal of Remote Sensing*, vol. 16, no. 5, pp. 877–889, 1995.

[90] J. E. Odegard, H. Guo, M. Lang, C. S. Burrus, R. O. Wells, Jr., L. M. Novak, and M. Hiett, "Wavelet based SAR speckle reduction and image compression," in *SPIE Proc. on Algorithms for Synthetic Aperture Radar Imagery II at AeroSense '95*, vol. 2487, Orlando, FL, pp. 259–271, 1995.

[91] T. Ojala, T. Mäenpää, M. Pietikäinen, J. Viertola, J. Kyllönen, and S. Huovinen, "Outex - new framework for empirical evaluation of texture analysis algorithms," in *16th International Conference on Pattern Recognition*, vol. 1, Quebec, Canada, 2002, pp. 701–706. [Online]. Available: http://www.outex.oulu.fi

[92] T. Ojala, M. Pietikäinen, and T. Mäenpää, "Multiresolution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971–987, 2002.

[93] C. J. Oliver, "Information from SAR images," *Journal of Physics D-Applied Physics*, vol. 24, pp. 1493–1514, 1991.

[94] ——, "Rain forest classification based on SAR texture," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 2, pp. 1095–1104, 2000.

[95] C. J. Oliver, I. McConnell, and D. Stewart, "Optimim texture segmentation of SAR clutter," in *Proc. of the EUSAR Conference (EUSAR 96)*, pp. 81–84, 1996.

[96] C. Oliver and S. Quegan, *Understanding Synthetic Aperture Radar Images*. Norwood: Artech House, 1998.

[97] R. A. Peters II, "A new algorithm for image noise reduction using mathmematical morphology," *IEEE Transactions on Image Processing*, vol. 4, no. 5, pp. 554–568, 1995.

[98] M. Porat and Y. Y. Zeevi, "Localized texture processing in vision: Analysis and synthesis in the gaborian space," *IEEE Transactions on Biomedical Engineering*, vol. 36, no. 1, pp. 115–129, 1989.

[99] L. J. Porcello, N. G. Massey, R. B. Innes, and J. M. Marks, "Speckle reduction in synthetic-aperture radars," *Journal of the Optical Society of America*, vol. 66, no. 11, pp. 1305–1311, 1976.

[100] R. Porter and N. Canagarajah, "A robust automatic clustering scheme for image segmentation using wavelets," *IEEE Transactions on Image Processing*, vol. 5, no. 4, pp. 662–665, 1996.

[101] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge, UK: Cambridge University Press, 1994.

[102] P. P. Raghu, R. Poongodi, and B. Yegnanarayanna, "Unsupervised texture classification using vector quantization and deterministic relaxation neural network," *IEEE Transactions on Image Processing*, vol. 6, no. 10, pp. 1376–1387, 1997.

[103] T. Randen and J. H. Husøy, "Multichannel filtering for image texture segmentation," *Optical Engineering*, vol. 33, no. 8, pp. 2617–2625, 1994.

[104] R. K. Raney and G. J. Wessels, "Spatial considerations in SAR speckle simulation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 26, no. 5, pp. 666–672, 1988.

[105] J. A. Richards, *Remote Sensing Digital Image Analysis An Introduction.* Berlin: Springer-Verlag, 1995.

[106] F. Safa and G. Flouzat, "Speckle removal on radar imagery based on mathematical morphology," *Signal Processing*, vol. 16, pp. 319–333, 1989.

[107] P. Salembier and J. Serra, "Flat zones filtering, connected operators, and filters by reconstruction," *IEEE Transactions on Image Processing*, vol. 4, no. 8, pp. 1153–1160, 1995.

[108] D. Schonfeld and J. Goutsias, "Optimal morphological pattern restoration from noisy binary images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 1, pp. 14–29, 1991.

[109] M. A. Schulze and Q. X. Wu, "Noise reduction in synthetic aperture radar imagery using a morphology-based nonlinear filter," in *Proc. Digital Image Computing: Techniques & Applications (DICTA 95)*, pp. 661–666, 1995.

[110] J. Serra, *Random Sets and Integral Geometrey.* London: Academic Press, 1982.

[111] K. Shanmugam, V. Narayanan, V. S. Frost, J. A. Stiles, and J. C. Holtzman, "Textural features for radar image analysis," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 19, no. 3, pp. 153–156, 1981.

[112] M. N. Shirazi, H. Noda, and N. Takao, "Texture modeling and classification in wavelet feature space," in *Proc. IEEE International Conference on Image Processing (ICIP 2000)*, vol. 1, pp. 272–275, 2000.

[113] M. E. Shokr, "Texture measures for sea-ice classification from radar images," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 89)*, vol. 1, pp. 113–116, 1989.

[114] M. S. Silverman, D. H. Grosof, R. L. De Valois, and S. D. Elfar, "Spatial-frequency organization in primate striate cortex," in *Proc. of the National Academy of Sciences USA*, vol. 86, pp. 711–715, 1989.

[115] M. Simard, G. DeGrandi, K. P. B. Thompson, and G. B. Benie, "Analysis of speckle noise contribution on wavelet decomposition of SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 6, pp. 1953–1962, 1998.

[116] M. Simard, S. S. Saatchi, and G. DeGrandi, "The use of decision tree and multiscale texture for classification of JERS-1 SAR data over tropical forest," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 38, no. 5, pp. 2310–2321, 2000.

[117] M. Simard, "Extraction of information and speckle noise reduction in SAR images using the wavelet transform," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 98)*, vol. 1, pp. 4–6, 1998.

[118] J. Sittigorn, Y. Rangsanseri, and P. Thitimajshima, "Incorporating spatial information into fuzzy clustering of multispectral images," in *Proc. of the Asian Conference on Remote Sensing (ACRS)*, 2002.

[119] K. Sivakumar and J. Goutsias, "Morphological size densities: Application to optimal texture classification and noise filtering," in *Proc. IEEE Workshop on Nonlinear Signal and Image Processing (NSIP 97)*, Michigan, USA, 1997.

[120] ——, "Morphologically constrained GRFs: Applications to texture synthesis and analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 2, pp. 99–113, 1999.

[121] J. Sklansky, "Image segmentation and feature extraction," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 8, no. 4, pp. 237–247, 1978.

[122] M. J. T. Smith and T. P. Barnwell, "Exact reconstruction techniques for tree-structured subband coders," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 34, no. 3, pp. 434–441, 1986.

[123] L.-K. Soh and C. Tsatsoulis, "Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 37, no. 2, pp. 780–795, 1999.

[124] P. Soille, *Morphological Image Analysis: Principles and Applications.* Berlin: Springer-Verlag, 1999.

[125] Y.-W. Song and S. S. Udpa, "A new morphological approach for reducing speckle noise in ultrasound images," in *Proc. 39th IEEE Midwest Symposium on Circuits & Systems*, vol. 3, pp. 1397–1400, 1996.

[126] M. Sonka, V. Hlavac, and R. Boyle, *Image processing, analysis and machine vision: 2nd edition.* Pacific Grove, CA: PWS, 1999.

[127] A. Sowter, G. Lemoine, and H. van Leeuwen, "SAR techniques for rapid area estimation of agricultural crops during winter and spring," in *IEE Colloquium on Integrated Systems for Commercial Remote Sensing Applications*, pp. 6/1–6/5, 1998.

[128] S. R. Sternberg, "Grayscale morphology," *Computer Vision, Graphics and Image Processing*, vol. 35, pp. 333–355, 1986.

[129] G. Strang and T. Nguyen, *Wavelets and Filter Banks.* Wellesley MA 02181: Wellesley-Cambridge Press, 1996.

[130] J. R. Sveinsson and J. A. Benediktsson, "Speckle reduction and enhancement of SAR images in the wavelet domain," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 96)*, vol. 1, pp. 63–66, 1996.

[131] R. Touzi, A. Lopes, and P. Bousquet, "A statistical and geometrical edge detector for SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 26, no. 6, pp. 764–773, 1988.

[132] P. M. Treitz, O. R. Filho, P. J. Howarth, and E. D. Soulis, "Textural processing of multi-polarization SAR for agricultural crop classification," in *Proc. IEEE*

International Geoscience and Remote Sensing Symposium (IGARSS 96), vol. 4, pp. 1986–1988, 1996.

[133] F. T. Ulaby, F. Kouyate, B. Brisco, and T. H. L. Williams, "Textural information in SAR images," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 24, no. 2, pp. 235–245, 1986.

[134] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Transactions on Image Processing*, vol. 4, no. 11, pp. 1549–1560, 1995.

[135] F. van der Heijden, *Image Based Measurement Systems Object Recognition and Parameter Estimation.* Chichester: John Wiley and Sons, 1994.

[136] L. Vincent, "Morphological area openings and closings for grey-scale images," in *Shape in Picture: Mathematical Description of Shape in Grey-level Images*, pp. 196–208, 1993.

[137] ——, "Morphological area openings and closings for greyscale images," in *Proc. NATO Shape in Picture Workshop.* Springer-Verlag, pp. 197–208, 1992.

[138] ——, "Grayscale area openings and closings, their efficient implementation and applications," in *Proc. EURASIP Workshop on Mathematical Morphology and its Applications to Signal Processing*, pp. 22–27, 1993.

[139] L. Vincent and P. Soille, "Watersheds in digital spaces: An efficient algorithm based on immersion simulations," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 583–598, 1991.

[140] K. D. Ward, "Compound representation of high resolution sea clutter," *Electronics Letters*, vol. 17, no. 16, pp. 561–563, 1981.

[141] T. P. Weldon, W. E. Higgins, and D. F. Dunn, "Gabor filter design for multiple texture segmentation," *Optical Engineering*, vol. 35, no. 10, pp. 2852–2863, 1996.

[142] Q. X. Wu and M. A. Schulze, "Statistical properties of three-look Jers-1 SAR data," in *Proc. of Image and Vision Computing New Zealand (IVCNZ)*, pp. 197–202, 1995.

[143] J. Xiao, J. Li, and A. Moody, "A detail-preserving and flexible adaptive filter for speckle suppression in SAR imagery," *International Journal of Remote Sensing*, vol. 24, no. 12, pp. 2451–2465, 2003.

[144] W. Xiaodan, J. Hua, and Z. Rongchun, "Texture segmentation method based on incomplete tree structured wavelet transform and fuzzy Kohonen clustering network," in *Proceedings of the 3rd World Congress on Intelligent Control and Automation*, pp. 2684–2687, 2000.

[145] Q. Yu, C. Moloney, and F. M. Williams, "SAR sea-ice texture classification using discrete wavelet transform based methods," in *Proc. IEEE International Geoscience and Remote Sensing Symposium (IGARSS 02)*, vol. 5, pp. 3041–3043, 2002.

[146] Y. Yu and S. T. Acton, "Polarimetric SAR image segmentation using texture partitioning and statistical analysis," in *Proc. IEEE International Conference on Image Processing (ICIP 2000)*, vol. 1, pp. 677–680, 2000.